3. FUNCTIONAL VIEW

The ICS Functional View describes the partitioning of ICS software elements, application level protocols used between the elements, and a discussion of how the elements might be arranged on physical hosts. Included is a Functional Framework which shows the arrangement of ICS elements and identifies the services provided by each element.

3.1 Architecture Foundations

This section discuss several topics which lay the ground work for the Functional Framework which is presented in the next section. This section discusses the conceptual design which was used during the requirements development; discusses the ICS as a generic three-tier architecture; and discusses how the choice of Z39.50 as the underlying protocol for CIP affects ICS.

3.1.1 ICS URD Physical Domain Model

The ICS URD [R2] describes ICS from a protocol-centric point of view. The ICS is defined around a common protocol, CIP, and the elements which speak CIP. The URD discusses the CIP domain as a virtual 'CIP space' within which CIP messages, consisting of requests and responses, are exchanged between architectural elements: clients, servers and middleware. The CIP space is bounded by interfaces to client-mappers and server-mappers, that is, software that converts CIP messages into external formats. (See also Section 2 for a discussion of CIP space.)

The URD [R2] also defines the ICS from a systems perspective in terms of providing a seamless interoperable environment whereby users can conveniently access all available data, whoever it belongs to, requiring both technical and administrative problems to be addressed by ICS. The ICS also must provide management of data access; user authentication, user session management, problems of routing (sending client requests to the correct server) and collation (accumulation of different sets of results). This leads to the concept of middleware, which sits between the client(s) and server(s).

The URD also comments that dividing the functionality between client, middleware, and CIP specific components of the server, is not to be resolved at the level of the URD. A range of architectural solutions can be envisaged, with different distributions of functionality and emphasis, and these should be traded off and evaluated at the level of system design analysis. These issues are addressed within this SDD.

3.1.2 Three Tier Architecture

The CIP space approach is a prime example of a generalized concept for distributed information systems which is referred to as the Three-Tier interoperability architecture (see Figure 3-1). The three-tiered application model splits an application into its three logical component types, clients, middleware, and data providers. The three tiers refer to the three logical component parts of an application, not to the number of machines used by the application. There may be any number of each of the component types within an application. Application components can be shared by any number of application systems. The application components communicate with each other using a distributed computing infrastructure. The three-tier architecture supports provider sites which manage their own resources and processes.

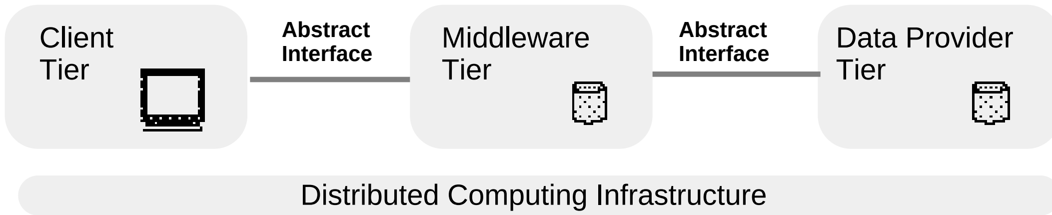


Figure 3-1. Three Tier Architecture

The Client tier consists of the user environment. A client component contains the logic which presents information to an external source and obtains input from that source. In most cases the external source is a human end user working at their own computer, although the external source might also be process-oriented. The client logic generally provides menus of options to allow the user to navigate through the different parts of the application, and it manipulates the input and output fields on the display device. Frequently, the presentation component also performs a limited amount of input data validation.

Middleware applications are entities that are used to model "business" processes. For ICS the "business" is catalogue interoperability. ICS middleware models the data and processes necessary for providing the client transparent access to heterogeneous catalogues at physically distributed locations. This is accomplished through interaction with the communication structure and an abstract interface, i.e., CIP.

Data providers contain the logic which interfaces either with a data storage system, or with some other type of external data source such as a data feed or an external application system. Data provider functions are generally invoked by the middleware, although in some applications they may be invoked directly by a presentation component.

A common mistake made in planning for three-tier architectures is assuming that, because a client/server application has three logical tiers, it must be implemented in three physical tiers. This misconception is unfortunate because it limits opportunities to capture application-specific requirements in the areas of user platform, network support, platform price/performance, development tools, and management capabilities. The three logical components can be distributed in many different ways to provide optimum configurations for application maintenance and support. The key is that once the three-tier architecture has been embraced in the design of an application, the partitioning of that application into its physical instantiation should be done in a way that optimizes performance, security, integrity, maintainability, and management.

Distributed computing infrastructures for three tier architectures is a current topic of much research and development that promises to provide transparent, dynamic access to services. For this purpose ICS relies instead on a proven and deployed infrastructure of Internet services, including DNS directory service. These are described in Section 5.

3.1.3 Z39.50 as the Base Protocol

Communications using the CIP are stateful, client/server sessions based on the Z39.50 Protocol [R9]. Z39.50 is an ANSI standard for information Retrieval. The Z39.50 protocol was chosen based on analysis of the URD requirements versus the services already defined in the Z39.50-1995 specification [R10]. CIP uses a subset of the Z39.50 messages to define an EO profile. CIP provides a set of EO specific attributes, as well as an extension of Z39.50 messages, through the provision of search control commands particular to the *collections* which are defined for ICS. CIP is the interface protocol for communications between all major components in ICS.

3.2 ICS Functional Framework

The ICS Functional Framework provides the top-level design of the application processes which make up ICS. In this section, the context of ICS is described first. Next, the ICS Elements are introduced according to their role in the three-tier architecture introduced in the previous section. Lastly the ICS Functional Framework which shows several sites is presented.

3.2.1 Context of ICS

The context of ICS is defined by the interfaces to systems external to ICS. Graphically this is displayed in Figure 3-2. (Note that in the convention of context diagrams, interfaces between external items are not shown, e.g., no interface is shown between Users and CEOS Agency Systems, as these are out of the scope of ICS.) There are four types of external systems with which ICS has interfaces: *Users, CEOS Agency Systems, Other CIP Based Federations, and Other Z39.50 Based Services.*

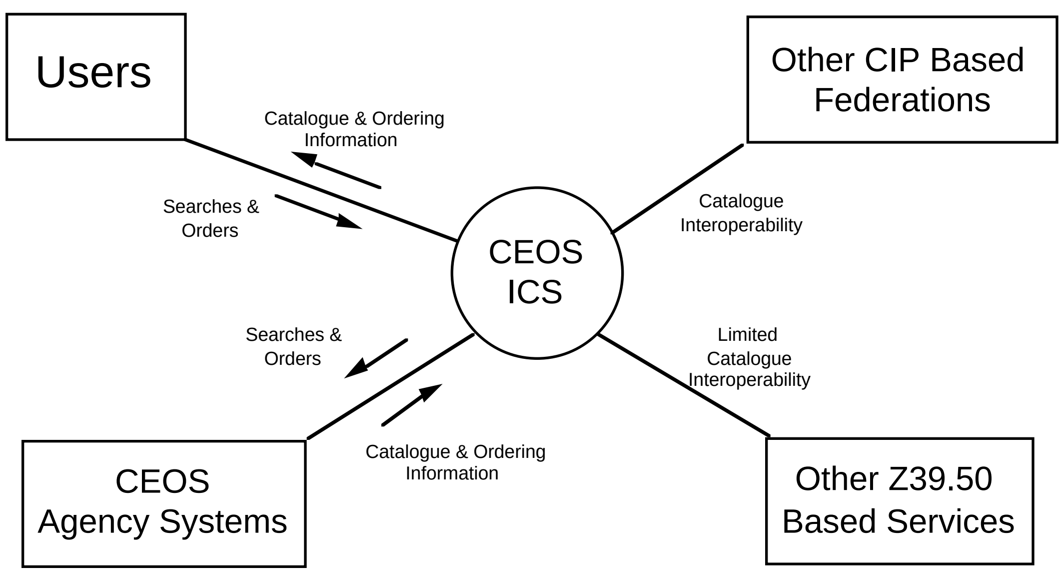


Figure 3-2 ICS Context Diagram

Users in Figure 3-2 are the human users which interface with ICS using a variety of clients. *Users* submit searches and orders to ICS and ICS responds with catalogue and ordering information respectively.

CEOS Agency Systems in Figure 3-2, are those data providers which provide catalogue services to ICS. When searches and orders are appropriately routed to the *CEOS Agency Systems,* they will respond with catalogue and ordering information respectively. *CEOS Agency Systems* includes affiliate CEOS agencies as well. Services assumed to be part of the *CEOS Agency Systems* are a catalogue service, order handling, user profiling, archiving, and site system management services.

Other CIP Based Federations in Figure 3-2, are groups of other data providers which provide catalogue interoperability using CIP. These other federations, may or may not have a design similar to ICS.

Other Z39.50 Based Services in Figure 3-2, are other groups of data providers which provide interoperability using Z39.50 but not necessarily using CIP. Catalogue interoperability with these providers will be limited to those supported by the specific Z39.50 profile which the providers is using.

3.2.2 Introduction of ICS Elements

Before describing how the ICS elements are connected to make a system, the individual elements are introduced. The elements are organized following the three tier architecture introduced earlier: Clients, Middleware and Data Providers

The main element of the client tier is the *CIP Client* (Figure 3-3). The *CIP Client* is used by a human user to compose searches and orders which are directed to the middleware elements. The *CIP Client* consists of presentation layer, a local management layer, and an application layer. The *presentation layer* deals with how the information is presented to the user, including all issues related to HMI, as well as dealing with certain format specific issues, e.g., displaying browse imagery. The *local management layer* provides functionally for local data management, e.g. saving a result set, or converting result sets into orders. The *application layer* deals directly with CIP including creating CIP messages and includes off-the-shelf Z39.50 communication software. Other configurations of clients derive from this basic set. For example, a web based client interfaces with a web server and *HTTP/CIP Gateway*, which incorporates the *Local User Management* and *Application Layers* of the *CIP Client*.

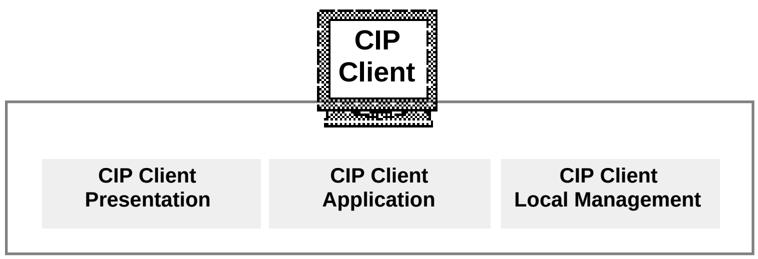


Figure 3-3 CIP Client

ICS elements which comprise the middleware tier are the *Retrieval Manager* and a series of *Translators*.

The *Retrieval Manager* is the main element of ICS (Figure 3-4). The *Retrieval Manager* holds the *collections* which are key to the distributed searching. The *Retrieval Manager* is key as it is the common element which routes the CIP messages. Furthermore the *Retrieval Manager* enables the following key ICS services:

- dynamic configuration of clients with semantic attributes
- *collection* searching
- routing to distributed sites
- access to diverse and heterogeneous catalogues
- flexible extensions and incorporation of additional services

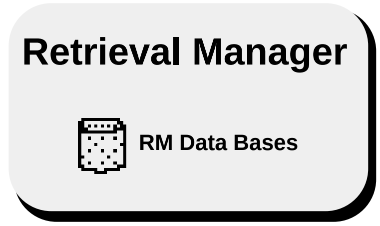


Figure 3-4 Retrieval Manager

Besides *Retrieval Managers,* the other type of middleware are the *translators* (See Figure 3-5). The *translators* provide the *Retrieval Manager* with the access points to the data providers. The *translators* convert CIP messages into the local messages of the data provider. Half of a *translator* is common across ICS, namely the portion of the *Translator* which handles CIP. The back half of a *Translator* is unique to each data provider and the internal protocols that are supported by data provider.

Catalogue functions as defined in the ICS URD includes the services of directory, inventory, guide, browse, and order. Three varieties of ICS *translators* have been defined. These three correspond to interactions with the data providers *Catalogue, Order Handling System (OHS) and User Profile System (UPS)*. The *Catalogue Translator* handles directory, inventory, guide and browse services. The *Order Translator* handles order services. Other *translators* may be required based on the partitioning of the functions provided by the data providers, e.g., a browse *Translator* would interact directly with a browse archive of a data provider. It is interesting to note that when a data provider uses CIP as the internal protocol, *Translator*s are not required.

Figure 3-5 ICS Translators

The last tier of the three tier architecture includes the data provider elements (Figure 3-6). Data provider elements are the elements which manage access and storage of the EO data of interest to the users. Strictly speaking the data provider elements are outside of the scope of ICS and are contained in the *CEOS Agency Systems* block on the ICS Context Diagram. Three data provider elements were introduced previously: *Catalogue*, *OHS*, and *UPS*. Additional elements are the *Archive* and the *Site System Management* (SSM). The *Translators* may not directly interact with the *archive* but for complete depiction of the ICS design, an *archive* holding the actual *product* data is needed. The *SSM* is present for those data providers which provide coordinated, on-line management of the distributed processing environment.

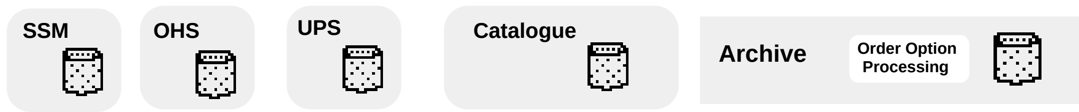


Figure 3-6 Data Provider Elements

There are other important elements which do not neatly fit into the three-tier architecture but are required for the ICS (Figure 3-7). The *Retrieval Manager Administrator (RMA)* is the human operator that performs all tasks needed to establish and maintain a *Retrieval Manager*. In practice this is more than one person as the tasks are of various types: a scientist for *collection* definition, a database expert for maintaining the *Retrieval Manager* databases, a system operator for diagnosis and correcting operational activities, etc. For convenience purposes all of these tasks are performed by the *RMA*.

The *Collection Management Tool (CMT)* (Figure 3-7), perform tasks involved with populating and maintaining the data in the Retrieval Manager. These tasks involve translating *collection* or directory information into CIP *collection* format and checking for valid entries. The *CMT* is used in conjunction with the data base administration tools which the *Retrieval Manager* provides.

The *Monitoring and Control Tools (MCT)* (Figure 3-7), provide the machine-to-machine interface for integrating the operations of the *Retrieval Manager* with the operations of a site. Through the *MCT*, the monitor and control the operations of the *Retrieval Manager* become part of the *SSM* operations for the data provider. (The *MCT* has been identified as an element in the ICS but specific functionality will be detailed in future releases of the SDD.)

Figure 3-7 Other ICS Elements

In addition to the elements described above, the ICS is hosted on a variety of hardware platforms and operates over a variety of networks. These issues are considered in other sections of the SDD, e.g. Sections 5 and 8.3.

3.2.3 Maximum ICS Site

This section defines how the ICS elements are configured into a data provider site. The elements were defined in the previous section. The site described here is a maximum site in that it contains all ICS elements (Figure 3-8). Particular sites have subsets of the elements. Variety amongst sites is addressed in the next section. The top half of the figure shows those elements which are strictly part of ICS. The bottom half of the figure shows those elements which exist within the *CEOS Agency Systems* and for the purposes of this discussion are labeled as ICS Related Elements.

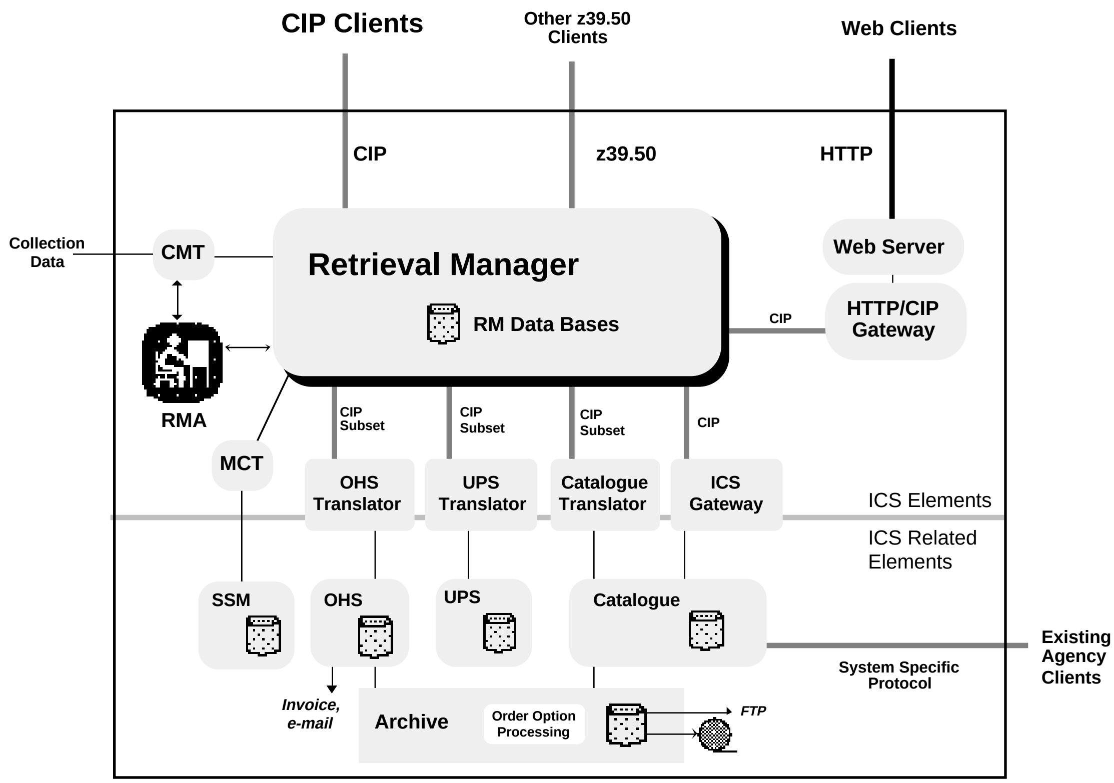


Figure 3-8 Maximum ICS Site

The heart of an ICS site is the *Retrieval Manager*. Virtually all ICS communication through the site passes through the *Retrieval Manager* and is communicated using CIP. A variety of Clients are shown interacting with the Retrieval Manager. A connection to a *CIP Client* is shown. WWW access is provided through a Web Server and *HTTP/CIP Gateway* interacting with the *Retrieval Manager*. The *Other Z39.50 Clients* are those clients which communicate using a profile of Z39.50, Version 3, but not CIP. These clients will be able to access some but generally not all of the *Retrieval Manager* services. An additional client is also shown in Figure 3-8, *Existing Agency Clients* will continue to access the site's catalogue. The site *catalogue* provides access to ICS through a *ICS Gateway*. This allows users of existing agency clients to perform searches and ordering of ICS items.

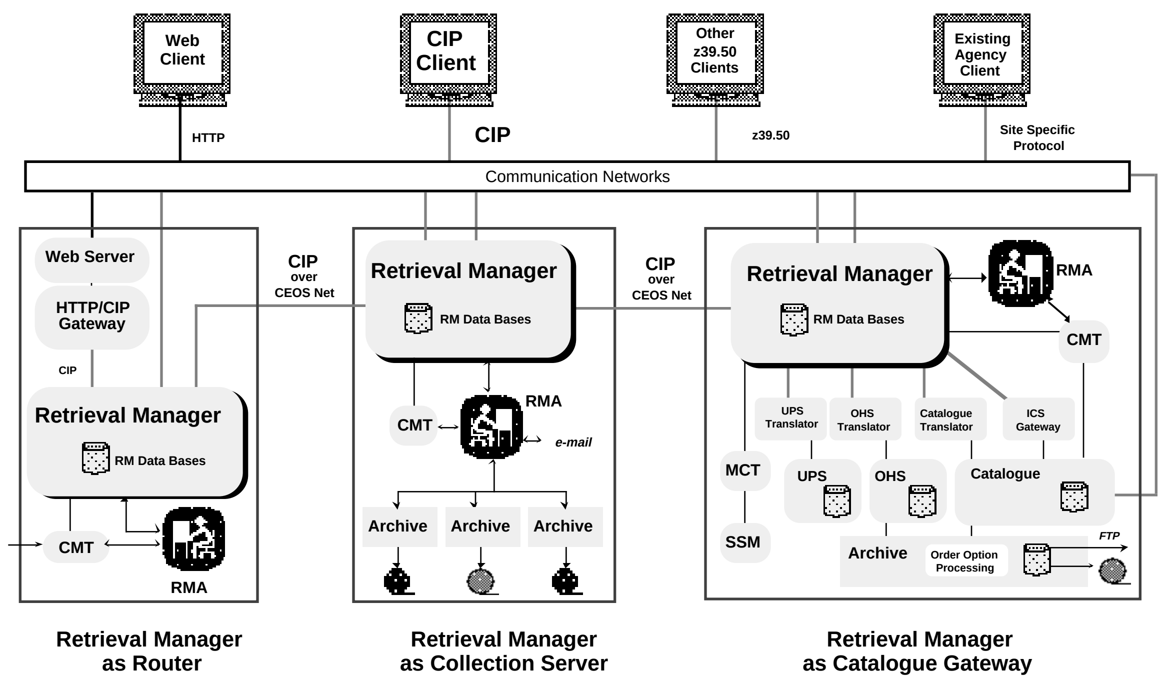
The *Retrieval Manager* interacts with a series of *Translators* to access the data providers services. Between the *Retrieval Manager* and the *Translators*, a subset of CIP messages particular to the *translator* are used for communications. Only a subset is supported because the *Retrieval Manager* routes queries to the *Catalogue* and orders to the *OHS*, therefore an *OHS* is not designed to respond to provide query responses. Between the *Translators* and the provider site services, the protocols for communication within the site are used. The data provider elements perform the necessary catalogue services or interact with other site resources, e.g., archive, to respond to the requests coming from the *Translators*.

Several other elements are shown in Figure 3-8 which pertain to keeping the ICS site operational. The *RMA* is shown interacting with the *Retrieval Manager* to establish and maintain the *Retrieval Manager*. This is done through a GUI provided by the *Retrieval Manager*. The *RMA* uses the *CMT* to populate and maintain the data in the *Retrieval Manager*. To support integration of *Retrieval Manager* operations with the *SSM*, an *MCT* is provided to monitor an control *Retrieval Manager* operations as directed by the *SSM*.

3.2.4 ICS as a System

Figure 3-9 shows the ICS Functional Framework. The framework shows a variety of arrangement of elements at ICS sites. Specific arrangement of ICS elements provided at ICS sites will vary over time. That is Clients, *Retrieval Managers* and even Agency holdings will be added, or taken away, dynamically, during the ICS operations. The ICS must enable this dynamic configuration. Therefore the ICS Framework shown in Figure 3-2 is only representative of how the ICS will be implemented at any given time.

The ICS Framework shows three sites which emphasis different roles at the sites. The three roles are describe in this section : *Retrieval Manager* as a Catalogue Gateway, *Retrieval Manager* as a Collection Server, *Retrieval Manager* as a Router.



CIP = Catalogue Interoperability Protocol
RMA = RM Administrator
OHS = Order Handling System
CMT = Collection Management Tools
RM = Retrieval Manager
UPS = User Profile System
MCT = Monitoring and Control Tools
SSM = Site System Management

Figure 3-9 ICS Functional Framework

Retrieval Manager as a Catalogue Gateway. For locations with existing catalogue and archive systems, the *Retrieval Manager* holds the *collections* which are particular to the ICS domain. The *product* metadata which is in the agency catalogue database and is accessed through a *Catalogue Translator* dynamically to satisfy queries. Likewise, the *Retrieval Manager* in this case holds user management data for only the ICS domain users (which will include the other *Retrieval Managers*). The *Retrieval Manager* interfaces to existing *User Profile Systems* and *Order Handling Systems* through *Translators*.

Retrieval Manager as a Collection Server. The middle configuration in Figure 3-9 shows how a *Retrieval Manager* is used to provide access to a more limited archive with minimal resources. In this case, the *Retrieval Manager* is a standalone system which provides the *collections*, *product* and *guide* searches. No *Translator*s are used. The *Retrieval Manager* contains the metadata for *collections*, *products*, and *guide*. The user then retrieves the actual *products* and *guides* via ftp, for example, from the archive. Ordering and authentication need not be implemented at this site. Setting up this form of a *Retrieval Manager* may be simplified by implementing the *Retrieval Manager* databases as a set of ASCII indexed files. Administration of the indexed files is accomplished through the *Retrieval Manager.*

Retrieval Manager as a Router. An additional variant of the *Retrieval Manager* implementation is an ICS router node in the ICS Framework (*Retrieval Manager* on the left side in Figure 3-9). A *Retrieval Manager* as a router functions as a "well known" ICS resource enabling users to find other *Retrieval Manager*s for specific data providers. An ICS Router is a middleware node containing *collections* but no *product* or *guide* metadata. For Release B, an ICS router will be used for the ICS Global Collection providing a "well known" resource to access all ICS *Retrieval Managers* (see Section 4 for discussion of Global Collection).

The Functional Framework, Figure 3-9, shows a *Retrieval Manager* interacting with only one *Catalogue Translator*. Other configurations for *Retrieval Manager*-to-*Translator* interfaces are supported. As shown in Figure 3-10, a *Retrieval Manager* can direct CIP messages to more than one *Catalogue Translator*. The converse is not true, namely a *translator* can not be served by more than one *Retrieval Manager*. Limiting the *Translator* in this fashion reduces the burden on the *translator* to manage CIP messages from more than one source, making the *translator* a simpler design. The purpose of the configuration of Figure 3-10 is to allow a *Retrieval Manager* to serve more than one *catalogue*. The multiple catalogues may be at different sites but under the same administration. Or a site may host a *translator* and *catalogue* and rely on a different site and organization to manage the *Retrieval Manager*. If a *catalogue* needs to provide access to multiple *Retrieval Managers*, multiple *translators* are provided as the existing *catalogue* is designed to handle multiple clients.

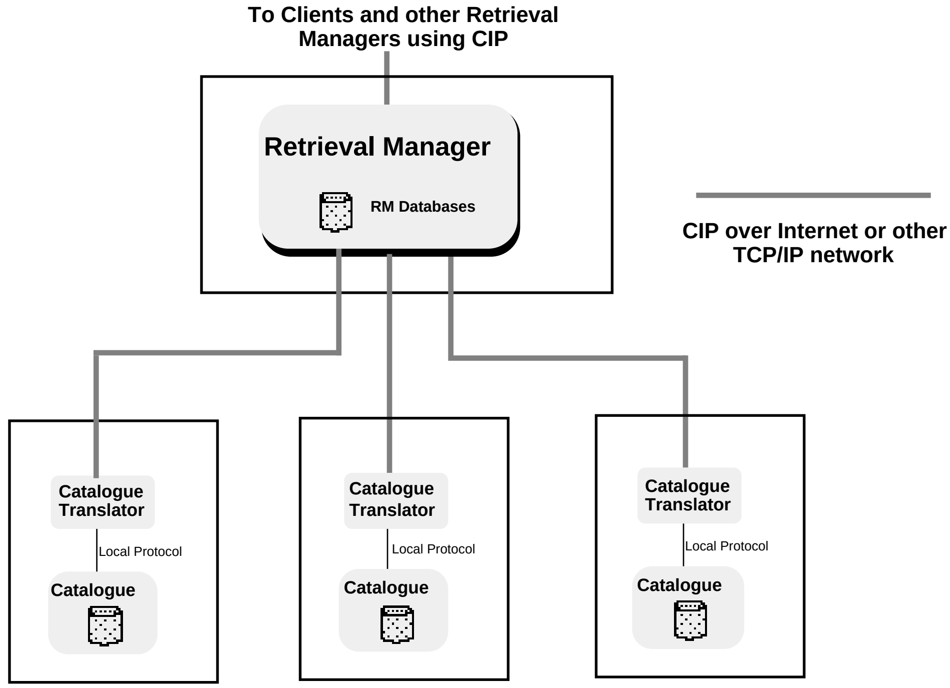


Figure 3-10. Multiple Translators Serviced by a Retrieval Manager

3.3 Catalogue Interoperability Protocol (CIP)

The Catalogue Interoperability Protocol (CIP) is the main method by which the objective of catalogue interoperability is achieved in ICS. For the ICS, CIP is the main interface between elements. CIP is elaborated in the CIP Specification [R3]. The CIP Specification defines the attributes which are specific to EO data catalogues and common across *Retrieval Managers* and *Translators* which speak CIP. An overview of the CIP messages and services are listed below to allow an understanding of the role CIP plays in ICS.

CIP messages are exchanged between an *origin* and a *target* as part of a *CIP Session*. An *origin* initiates a session with a *target*. Several ICS elements act as *origins*: *CIP Client, Retrieval Manager, ICS Gateway, and HTTP/CIP Gateway*. Several ICS elements act as *targets*: *Retrieval Managers* and *Translators.*

The messages which CIP uses are listed Table 3-1. The messages are used in the SDD to demonstrated the use of CIP in ICS. The CIP Specification [R3] provides a controlled definition of the contents and meaning of each of the messages in Table 3-1.

Table 3-1. CIP Messages

CIP Message Name	Short Description of CIP Message
InitializeRequest	Used by an *origin* to establish a *session* with a *target.*
InitializeResponse	Generated by the *target* after reception of the *InitializeRequest*
SearchRequest	Defines the elements transferred for any search operation including the query, the target database, and a result set name
SearchResponse	Contains the response to a *SearchRequest*, in particular it contains the total number of records in the result set
PresentRequest	Specifies the characteristics of the records in a result set which are to be returned to the *origin*
PresentResponse	Return of the records requested during a *PresentRequest*.
SegmentRequest	When the records requested by a *PresentRequest* will not fit in a single segment , the *target* returns multiple *SegmentRequest*, each containing a portion of the requested records.
DeleteResultSetRequest	Allows a *target* to request an *origin* to delete specified result sets created during a *CIP session*
DeleteResultSetResponse	Generated by the *target* in response to the *DeleteResultSetResponse*
AccessControlRequest	Allows a *target* to challenge the identity of an *origin* .
AccessControlResponse	Sent by the *origin* to the *target* in response to a “challenge” to the identity of the *origin* through an AccessControlRequest
ResourceControlRequest	Allows the *target* to provide information concerning the status of a query to the *origin*.
ResourceControlResponse	Allows the *origin* to indicate whether to continue a query or not.
TriggerResourceControlRequest	Used by the *origin* to request a status report or change the status of an existing query. May be used to cancel a query started by the *origin* .
ResourceReportRequest	Allows an *origin* to request a *target* to produce a resource report.
ResourceReportResponse	Generated by the *target* in response to the *ResourceReportRequest* .
ExtendedServicesRequest	Allows an *origin* to perform tasks which are executed outside the scope of Z39.50. CIP supports the following extended services: Persistent Result Sets, Persistent Query, Periodic Query Schedule , Database Update, CIP Ordering The CIP Ordering Extended Service can contain one of the following actions: *orderQuoteAndValidate*, *orderSubmit*, *orderMonitor*, and *orderCancel*.
ExtendedServicesResponse	Allows a *target* to send back the response related to the execution of tasks executed outside the scope of Z39.50 and initiated by an *ExtendedServicesResponse*.
Close	Allows an *origin* or *target* to abruptly terminate all active operations and to initiate the termination of the *CIP Session*

Session Initialization and Closure. CIP is a session based protocol. A *CIP session* begins with an initialization *operation* and ends with a closure *operation*. The initialization allows negotiation procedures to control the use of all services to be used during the session. The closure *operation* signals the end of the session and is done unilaterally by either the *origin* or *target*.

Search. The Search service is the means by which the *origin* sends a query to be evaluated against the EO data and the status of the query is returned. A CIP Search consists of a number of fundamental parts:

- Search control parameters
- The search query using a Z39.50 language and containing a valid combination of CIP attributes
- The target database for the search
- The result set created at the *target* that holds the results of the search

Search Control. Search control parameters allow the user to control the type of search requested. The user can select either a *collection* search, a *product* search, or a *guide* search. A *collection* search is a CIP search that is used to identify and retrieve *collection* definitions. A *product* descriptor search is a CIP search that is used to identify and retrieve *product* descriptors. A *guide* search is used to identify *guide* documents of interest. Also, the user can select any search to be either local to the *Retrieval Manager* which holds the targeted *collection*, or allow the search to be distributed to other *Retrieval Manager*s as identified in the searched *collection*.

Query Language. CIP defines queries using a general purpose query language defined using Reverse-Polish-Notation (RPN). Note, however, that this does not preclude *Retrieval Manager*s to support other additional Z39.50 query languages.

Search Attributes. CIP splits attributes into different types depending upon their function: Use, Relation, Position, Structure, Truncation and Completeness attributes. In a search query, the Use attribute identifies the access point against which the search term is to be matched. A set of Use attributes based on the EO domain are defined and controlled in the CIP Specification. The other types of attributes are used to provide additional match criteria in a query.

In addition to the CIP attributes defined in the CIP specification, each data provider may define its own local attributes. These local attributes are used in exactly the same way as the CIP attributes. The only restriction is that their understanding is limited to the domain of the data provider who defines them (instead of the whole CIP domain as for the CIP attributes). For instance, if a search query contains a local attribute, this local attribute will be applicable (i.e. will be recognized) only by the *collections* owned by the appropriate *Retrieval Manager*.

Result Set Retrieval. Using a *presentRequest*, CIP returns records that have been located through execution of the search service. Once a CIP *searchRequest* has been submitted to the *Retrieval Manager*, performed at the *target*(s), and has responded successfully, a result set is made available for subsequent *presentRequests* by the *origin*. Optionally, an *origin* may request piggybacking in a CIP search request, in which case a small number of retrieval records are returned in the search response itself. The retrieval record formats for piggybacked records are the same as those described for present responses. By selecting from pre-defined element sets, the user can select the desired fields to be returned from the result set.

Extended Services. *Extended Services* provide a mechanism to define and monitor tasks that are executed outside Z39.50. An *Extended Service* (ES) defines a particular task which is related to information retrieval but is not defined within Z39.50. It allows an *origin* to create, modify or delete *task packages*, which are maintained by the *target* in a database - the *Extended Services database*. The task defined in a *task package* depends on the particular *Extended Service* used. For CIP the following *Extended Services* are provided: Persistent Result Sets, Persistent Query, Periodic Query Schedule, Database Update, and CIP Ordering. Note that an *ES response* does not necessarily signal the completion of a task, which may have a lifetime which exceeds the *CIP Session* during which it is initiated.

Ordering: CIP defines an *Extended Service* for ordering of EO *products*. CIP supports submitting an order specification for validation and to obtain a quote. A specific message is provide for affirmative submission of an order. Additionally, a user can monitor the status of a order and request to cancel the order.

Authentication: CIP provides the capability to transfer authentication information between *target* and *origin* pairs of a *CIP Session*. This transfer allows *Retrieval Managers* to authenticate *CIP clients* and other *Retrieval Managers*. The authentication may be done at the beginning of a *CIP Session* or may be requested by a *Retrieval Manager* for a specific *operation*. Additionally, a *Retrieval Manager* can request non-repudiation be enforced for a specific request.

Explain. The Explain service enables the capabilities of a target server to be ascertained by clients. Explain queries are targeted at a *Retrieval Manager* and thus at Explain database. The Explain database contains the attributes which can be used at the *Retrieval Manager* as well as key information about the EO content of the *Retrieval Manager*.

Identifiers. The CIP Specification also defines syntax for identifiers of several key items in the CIP space: messages and *collections* (including result sets). For example, each operation request from a client (or *Retrieval Manager*) must have a unique reference identifier. This can be used in the tracking of an operation and is particularly important when a search is invoked which is passed onto remote *Retrieval Manager*s.

Unsupported Operations. Not all ICS elements will support all CIP operations. Negotiation of supported operations occurs during *session* initialization. ICS elements as part of initialization will identify the operations supported by an element. If a message for an unsupported operaitons is sent during the session, e.g., a CIP ordering message is sent to a Catalogue Translator, the ICS elemetn shall respond to the unsupported message with a dianostic message. The diagnostic message will be code 100 from the

3.4 CIP Operations

Messages are passed between ICS elements primarily using CIP. CIP is a session based protocol for conducting the operations, e.g., search and order. This section shows how various messages are passed between the distributed ICS elements to accomplish a request by a user for a query or an order. Each section discusses the simpler “local” case, and then discusses the case when multiple *Retrieval Managers* are involved. The operations shown here are examples of the two main interactions in ICS - search and order - and do not demonstrate the full extent of how ICS is used.

In each scenario it is assumed that a session is begun before the scenario is described. A scenario is established by the Client sending an *InitializeRequest* to the *Retrieval Manager* and the *Retrieval Manager* replying with an *InitializeResponse*. Likewise when a *Retrieval Manager* must send a secondary message to a another *Retrieval Manager* or to a *Translator*, a session is formed between the two. Nominally, the Client will send a *close* message to end the session. There is no response from the *Retrieval Manager* to the Client for a *close*. When the *Retrieval Manager* receives a *close* it will send a *close* to any other ICS elements to which a session had been established for this user.

3.4.1 Queries

When the user wants to find records pertaining to a particular interest, there will be multiple CIP messages behind the scenes which the CIP Client may or may not make visible. The CIP messages provide the flexibility to retrieve only the information that the user wishes to view. For instance a query consists of two separate operations: a search operation and a present operation. The search operation returns the number of hits. The present returns the records in the particular format requested by the user.

The following two sections show how the combination of search and present operations are used to search ICS data. First a local query is described followed by a distributed query. The particular examples shown in this section are typical but not the only combination of operations which a user may cause to happen in ICS. For example, a user query session may include multiple searches and only one present.

3.4.1.1 Local Query

Figure 3-11 shows the passing of messages between ICS elements to accomplish a local query. The query is searching the *collections* held by a local catalogue. The ICS elements involved are a *CIP Client Application*, a *Retrieval Manager*, and a *Catalogue Translator*. The client targets a *searchRequest* at a *collection* held by the *Retrieval Manager*. The contents of the *collection* in this case are held in the local catalogue and so the *Retrieval Manager* passes a *searchRequest* to the *Catalogue Translator*. The *searchResponse* which is returned from the Catalogue *Translator* and is passed to the client contains the number of "hits" produced by the search.

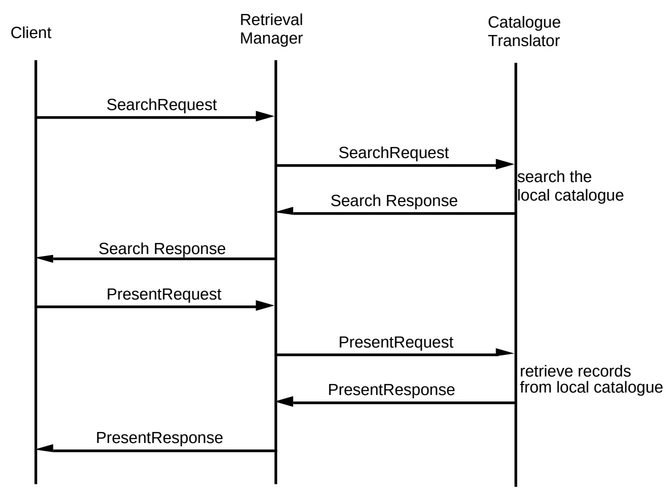


Figure 3-11 Local Query Operation Messages

Assuming that the user is satisfied by the number of hits indicated, the *CIP Client* is used to request the records in a specific format. This is accomplished by the client issuing a *presentRequest* against the result set which is held by the *Catalogue Translator*. The records are retrieved from the local catalogues and passed to the client via the *Retrieval Manager*.

The main function of the *Retrieval Manager* in this case has been to route messages to appropriate ICS elements based on the contents of the *collections*. Messages are routed based on what type of search was requested and where specific data is held. A user can request that a CIP search be either local to the *Retrieval Manager* or be propagated to other *Retrieval Managers* if the collection structure would warrant this for a specific query. The user may also choose between searching just the collections or searching the included products as well. If an ICS site holds the product descriptors in the local catalogue and not in the *Retrieval Manager*, the search will need to be forwarded to the catalogue if a Product Search was requested. How data should be distributed between the *Retrieval Manager* and the catalogue is dependent upon characteristics of an agency's data, e.g. number of products and volatility of *collections*. Each agency will determine the distribution of data while considering the aspects of maintaining replicated data versus the response time for processing a user request. These issues can be assessed at the time of configuring an ICS site. The CEOS PTT plans to provide guidance on these issues in the ICS Collections Manual [R5].

3.4.1.2 Distributed Query

The distributed query capabilities of the ICS are depicted by the messages in Figure 3-12. Similar to the previous section, the following operations are handled in a *CIP session* which is assumed to be established prior to the *operations* shown here.

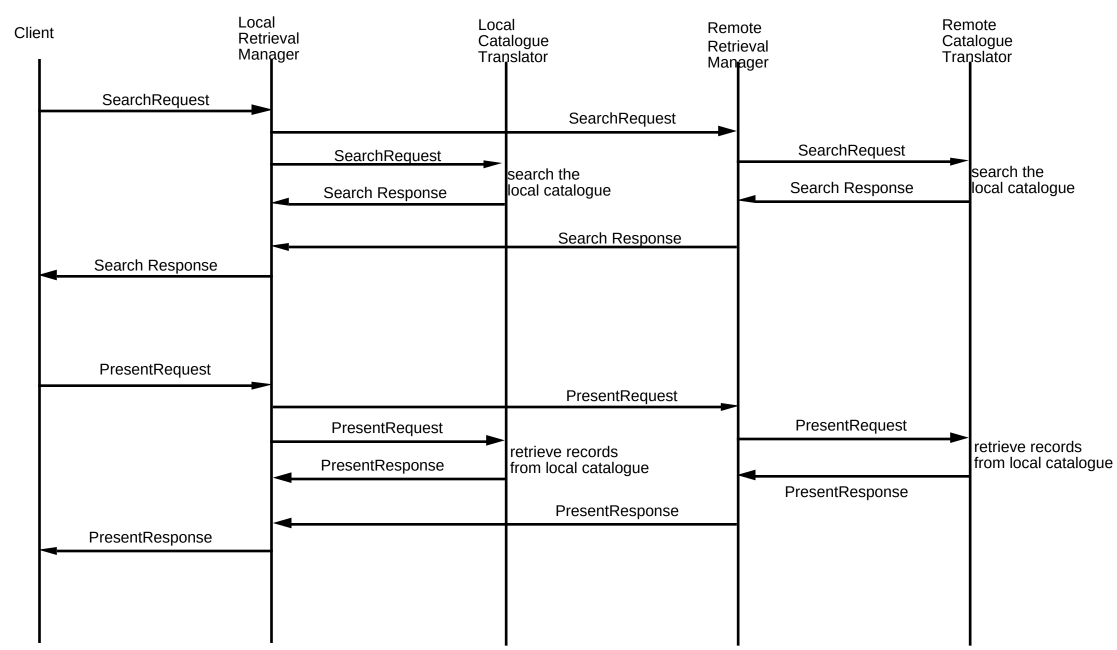


Figure 3-12 Distributed Query Operation Messages

When the *CIP Client* targets a *searchRequest* at a *collection* which includes both local *collections* and remote *collections*, the *Retrieval Manager* creates sub-searches which are targeted at the included *collections*. In this particular example, two separate *searchRequests* are created: one is sent to the local *Catalogue Translator*, the second is sent to a remote *Retrieval Manager*. The remote *Retrieval Manager* in turn determines, based on the *collection* contents, that a *searchRequest* needs to be sent to a *Catalogue Translator* which is local to the remote *Retrieval Manager*. *SearchResponses* are routed back to the initial *Retrieval Manager* which combines them, and a single *searchResponse* is sent to the client.

Similar to the splitting of the *searchRequest*, a *presentRequest* will be split into sub-requests based on the contents of the result set. The *presentRequests* are routed appropriately and *presentResponses* are returned to the initial *Retrieval Manager*. The responses are then combined and provided to the Client as a single *presentResponse*.

3.4.2 Ordering of EO Products

This section describes ordering using CIP. The ordering process is shown here as fundamentally a two step process. First, a step of submitting an order specification resulting in a quote. Second, the order is submitted for processing. There are multiple variations on this two step process. There may be multiple quotes requested before the user is happy with the order specification and then the order is submitted. On the other extreme the user may know exactly what they want and will directly submit the order without first asking for a quote. A user can also monitor and cancel an order, but these operations are shown in these scenarios.

This section, like the last previous section, first shows a single *Retrieval Manager* case followed by the case in which two *Retrieval Manager*s are involved. There is a key difference between queries and orders. A query may be distributed to multiple *Retrieval Manager*s and *Catalogues*. Orders must contain *products* from a single *OHS*. When the order is composed by the client it will contain requests for items from only one site. The order may be routed from the local *Retrieval Manager* to a remote *Retrieval Manager*, but the order will only be presented to one *OHS*. For this reason we distinguish between as “distributed” queries vs. “indirect” ordering.

Critical to ordering for some data providers will be the security mechanisms provided by CIP. Data providers will restrict some *products* to particular individuals and a monetary charge will be required for some *products*. CIP provides an authentication mechanism allowing a data provider to verify the identify of the person requesting a product. CIP also provides non-repudiation, which when invoked requires the user to submit an order which can not be disavowed at a later time.

3.4.2.1 Direct Ordering

The direct ordering scenario by a user involves a *primary order*, from the *CIP Client* to the *Retrieval Manager*, and a *secondary order* to the local OHS *Translator* (See Figure 3-13). A *primary order* is defined to be between a *CIP Client* and a *Retrieval Manager*. A *secondary order* is created by an *Retrieval Manager* in response to a *primary order* and may be either to another *Retrieval Manager* or to an *OHS Translator*. Like in the previous scenarios, the messages shown in this scenario occur as part of a *CIP Session* for which the initialization *operation* is not shown in the scenario. After the user has specified the contents of the order, the client sends a CIP Ordering *ExtendedServicesRequest*, with the action of *orderQuoteAndValidate,* to the *Retrieval Manager*. To support the ordering *Extended Service*, the *Retrieval Manager* creates a *Task Package* which contains the order as well as additional information related to the *operation*. The *Task Package* will be updated based on the additional *Extended Services* messages as the scenario proceeds. The *Retrieval Manager* determines which OHS can fulfill the specified order. In this case, the *extendedServicesRequest* is routed by the *Retrieval Manager* to a local *OHS Translator*. The *OHS* provides the *Retrieval Manager* with a quote in the *ExtendedServicesResponse* which is returned to the client.

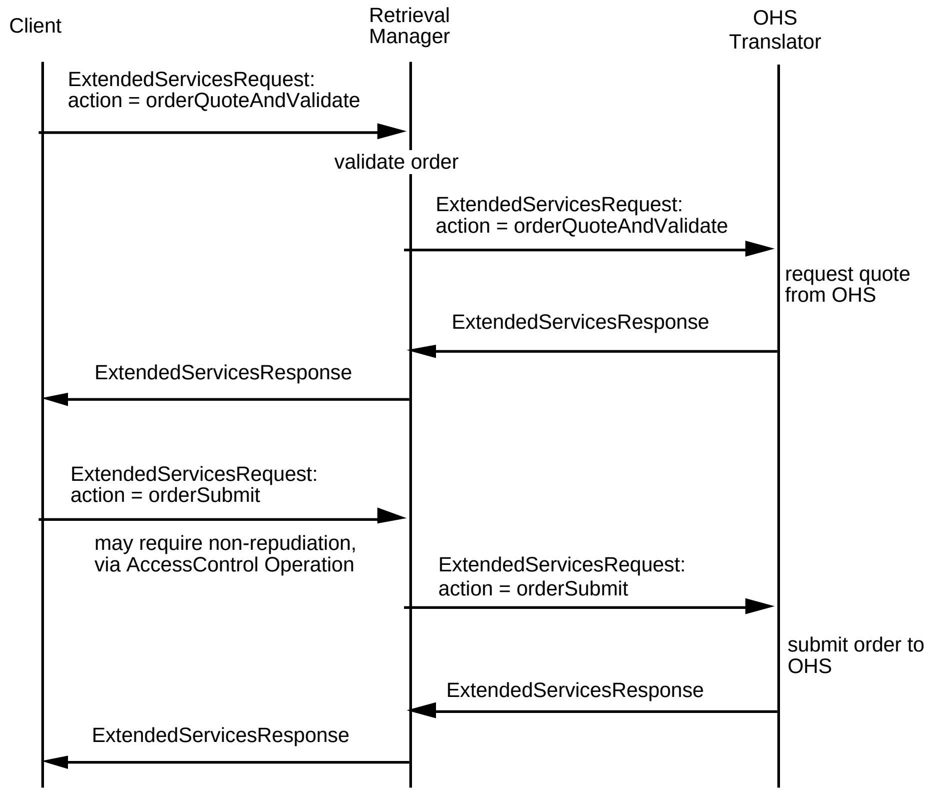


Figure 3-13 Direct Ordering

The scenario shown in Figure 3-13 continues with an order submittal. A user sends a CIP Ordering *extendedServicesRequest,* with the action of *orderSubmit,* when they are committing to the order. In some cases a *Retrieval Manager* will request that the order be submitted in a message which cannot later be denied, i.e. non-repudiation is required on the *orderSubmit*. Non -repudiation requires a *AccessControlRequest* message from the *Retrieval Manager* to the client and an *AccessControlResponse* message back from the client (not shown in Figure 3-13). The *Retrieval Manager* then submits the order using the *extendedServicesRequest* to the *OHS Translator*. An *extendedServicesResponse* is returned to the *Retrieval Manager* which in turn sends an *extendedServicesResponse* to the client.

3.4.2.2 Indirect Ordering

An indirect ordering scenario by a user involves a *primary order*, a *secondary order* between a local *Retrieval Manager* and a remote *Retrieval Manager*, and another secondary order between the remote *Retrieval Manager* and the OHS *Translator*.

An indirect ordering scenario may have the local *Retrieval Manager* acting as either a proxy or a passthrough. In the proxy case, access rights of the Local *Retrieval Manager* on the Remote *Retrieval Manager* are used to order from the Remote *Retrieval Manager.* This allows a user to order *products* when the user has no privileges with the Remote *Retrieval Manager*. In the pass-through case, the Local *Retrieval Manager* routes the information to the remote *Retrieval Manager*, but the order is requested using the privileges of the user.

The messages passed for an Indirect Order (Figure 3-14) are very similar to the direct ordering case. The major difference is that when the client submits an *extendedServicesResponse* with action of either *orderQuoteAndValidate* or *orderSubmit* to the *Retrieval Manager*, the *Retrieval Manager* identifies that the order is to be filled by an *OHS* to which it does not have direct access. The *Retrieval Manager* then becomes an intermediary and passes the requests off to a Remote *Retrieval Manager*. Both the local and the remote *Retrieval Manager* will create a task package associated with the order request that it receives from the client or local *Retrieval Manager*, respectively.

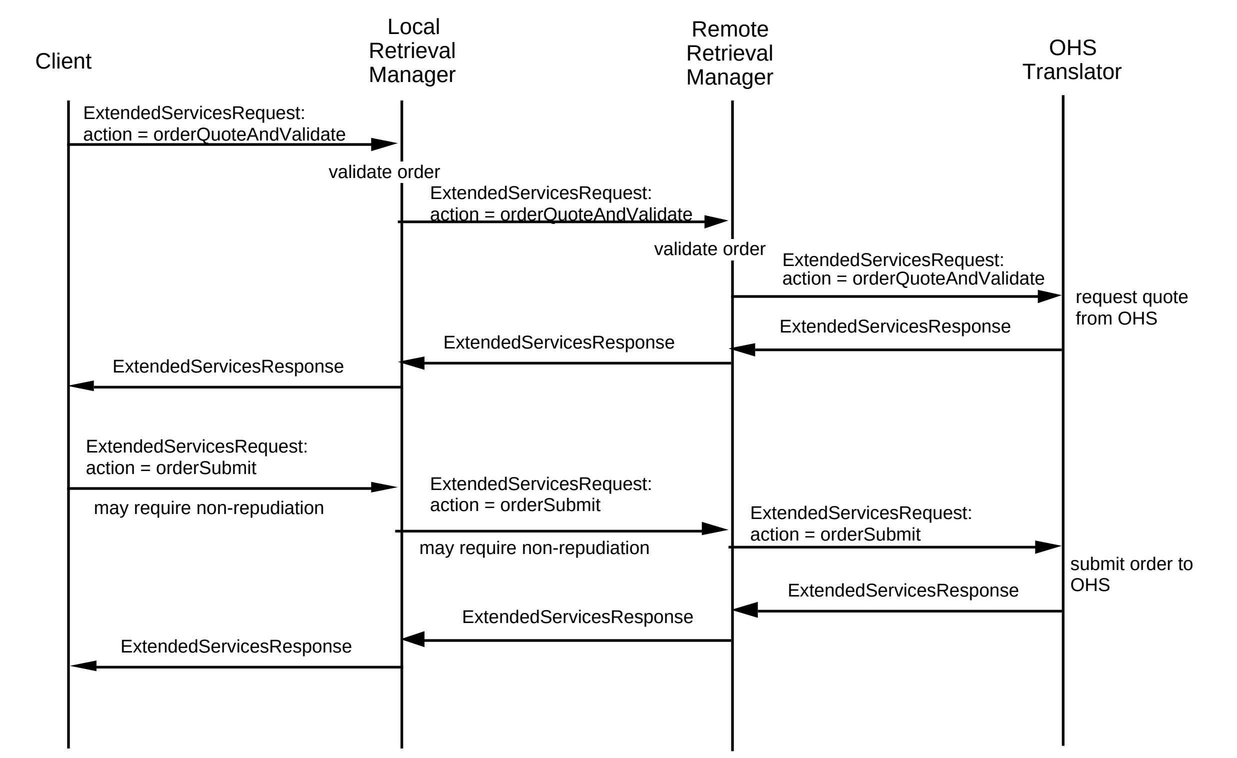


Figure 3-14. Indirect Ordering

3.5 Identification of ICS Element Services and Interfaces

This section provides an identification of the services provided by each element shown in the ICS Framework. The remainder of this section provides the identification of services for the following elements:

- *Retrieval Manager*
- *CIP Client Application*
- *Catalogue Translator Services*

- *OHS Translator Services*
- *UPS Translator Services*
- *HTTP/CIP Gateway*
- *ICS Gateway*
- *Retrieval Manager Administrator (RMA)*
- *Collection Management Tools (CMT)*
- *Monitoring and Control Tools (MCT)*

Section 3-6 describes services assumed to be provided by the non-ICS elements.

The services supplied by a given site, using ICS elements, will vary by site. The services in this section are defined assuming a maximum site. Variation across sites is described in section 9 by describing three cannonical ICS sites. Handling of unsupported CIP operations by an element at a given site shall be as described in 3.3.

3.5.1 Retrieval Manager Services

The following services are provided by the *Retrieval Manager*:

- CIP Compatible Messaging based on Z39.50 messaging and session management: *origin* and *target*
- Session Management: Manages user requests including; queries and associated sub-queries, search routing, recursive search trapping, result set collation and presenting, order management.
- Collection Management: persistent storage of local *collection*; services to create, ingest and maintain local *collection*; attribute management; promotion of hot *collections* to theme *collections*.
- Explain Database: Persistent Store of Explain data
- Extended Services: Creation, modification and persistent storage of *Task Package* in response to *Extended Service* requests and responses.
- User Management: persistent storage of user information where user may be a human using a client or may be another *Retrieval Manager*.
- Operator Interface for *Retrieval Manager Administrator*
- Monitoring and Error Management: User session errors, CIP diagnostic messages, network diagnostics messages
- E-mail Client: E-mail to *RMA* for error messages.

The *Retrieval Manager* design is required to be a modular design allowing a site to disable specific services which are not be hosted at the site. The variation of *Retrieval Manager* services across sites is described in Section 9.

The *Retrieval Manager* has the functional interfaces indicated in Table 3-2.

Table 3-2. Retrieval Manager Interfaces(1 of 2)

Other ICS or Related Element	**Interface Description**
CIP Client Application	A *Retrieval Manager* is capable of forming *CIP sessions* with multiple *CIP Clients* where the *CIP Client* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Other z39.50 Clients	*Retrieval Manager* is capable of forming a z39.50, Version 3, Z-association with other z39.50 clients, where the *Other Z39.50 Client* is the *origin* and *Retrieval Manager* is the *target*. All Z39.50, Version 3 messages are supported by this interface with the exception of the following z39.50-1995 services: *scanRequest, scanResponse, sortRequest, sortResponse, extendedServicesRequest, extendedServicesResponse.*
Other *Retrieval Manager*s	A *Retrieval Manager* is capable of forming CIP sessions with other *Retrieval Manager*s where either *Retrieval Manager* may be the *origin* or the *target*. All CIP messages are supported by this interface.
ICS Gateway	A *Retrieval Manager* is capable of forming *CIP sessions* with multiple *ICS Gateways* where the *ICS Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Catalogue Translator	A *Retrieval Manager* is capable of forming CIP sessions with multiple Catalogue *Translator*s where the *Retrieval Manager* is the *origin* and the Catalogue *Translator* is the *target*. All CIP messages are supported by this interface. Extended Service types do not include CIP ordering. (Note that other interface approaches are under investigation, e.g., direct function call, for those sites with tight coupling between the Retrieval Manager and *Translator*.
OHS Translator	A *Retrieval Manager* is capable of forming CIP sessions with multiple OHS *Translator*s where the *Retrieval Manager* is the *origin* and the OHS *Translator* is the *target*. The following CIP messages are supported by this interface: *InitializeRequest, InitializeResponse, AccessControlRequest, AccessControlResponse, ExtendedServicesRequest, ExtendedServicesResponse, Close.* The only valid extended services type for this interface is CIP ordering

Table 3-2. Retrieval Manager Interfaces(2 of 2)

Other ICS or Related Element	Interface Description
UPS Translator	A *Retrieval Manager* may have an interface to a UPS *Translator* to request User Management Data including User Authentication Information. This interface will be a later release of CIP. A Retrieval Manager may have an interface to a Certification Authority to perform public key directory lookups. This interface should use the X.509 Lightweight Directory Access Protocol (LDAP) [R23].
Retrieval Manager Administrator (RMA)	The *Retrieval Manager* provides an operator interface to the *RMA*. The operator interface provides a graphical user interface to allow the *RMA* to monitor, control, diagnose, and maintain the operations of the *Retrieval Manager*.
Collection Management Tools	The *Retrieval Manager* allows the *CMT* to modify the EO content of the *Retrieval Manager Collection Data Base*
Monitoring and Control Tools	The *Retrieval Manager* allows an *MCT* interface to gather operation and error handling data from the *Retrieval Manager* and to responds to *MCT* commands for controlling the configuration and state of *Retrieval Manager* processes. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.5.2 CIP Client Application Services

The following services are provided by a *CIP Client Application*:

- CIP Compatible Messaging. The *CIP Client Application* is able to send and receive all CIP messages.
- Search Formulation. The *CIP Client Application* supports the user in formulating queries, e.g., queries of *collection*s and *products*, incremental queries, conversion of geographic inputs into CIP queries.
- Result Set handling. The *CIP Client Application* supports the user in viewing result sets, conversion of a result set to a hot *collection*.
- Order Formulation. The *CIP Client Application* supports the user in specifying and reviewing quotes on orders, including conversion of a multi-site result set into single site orders.
- Authentication Support. The *CIP Client Application* supports the user in providing the user’s credentials to form an authenticated session or to respond to an authentication request from the *Retrieval Manager*.
- Dynamic Configuration. The *CIP Client Application* supports the retrieval and caching of attribute sets from *Retrieval Managers*. The *CIP Client* provides the attributes in support of a user creating searches.
- User Profile Management. Either the *CIP Client Application* layer or the *CIP Client Local Management* layer may support the persistent storage of user profile data.

Note that the list above intentionally does not include presentation services to the user. See Section 3.2.2 for a description of other *CIP Client* layers.

The *CIP Client Application* has the functional interfaces indicated in Table 3-3.

Table 3-3 CIP Client Application Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	*CIP Client Application* is capable of forming CIP sessions with at least one *Retrieval Manager* where the CIP Client is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Interfaces outside of ICS	The *CIP Client Application* may have interfaces to the following ICS related elements: *CIP Client local management layer*,

3.5.3 Catalogue Translator Services

The following services are provided by the *Catalogue Translator*.

- mapping of CIP attributes to local attributes (i.e., for *product* and *guide*)
- mapping of CIP query elements to local query elements
- mapping of local result sets into CIP result sets
- use of the local *catalogue* communication protocol

The *Translator* contains elements which are common across ICS sites, e.g., those parts of the *Translator* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with the *Catalogue*.

The *Catalogue Translator* has the functional interfaces indicated in Table 3-4.

Table 3-4 Catalogue Translator Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	A Catalogue *Translator* is capable of being a CIP *target* for *CIP sessions* from a single *origin*. Typically the *origin* will be a *Retrieval Manager*. All CIP messages are supported by this interface. Extended Service types do not include CIP ordering. (Note that other interface approaches are under investigation, e.g., direct function call, for those sites with tight coupling between the *Retrieval Manager* and *translator*.
Interfaces outside of ICS	The *Catalogue Translator* interfaces to the following ICS related elements: *Catalogue*

3.5.4 OHS Translator Services

The following services are provided by the *OHS Translator*.

- Convert CIP order objects to local order handling system objects
- use of the local *catalogue* communication protocol

The *Translator* contains elements which are common across ICS sites, e.g., those parts of the *Translator* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with the *OHS*.

The *OHS Translator* has the functional interfaces indicated in Table 3-5.

Table 3-5 OHS Translator Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	An *OHS Translator* is capable of being a CIP *target* for one CIP session. Typically the *origin* will be a Retrieval Manager. The following CIP messages are supported by this interface: *InitializeRequest, InitializeResponse, AccessControlRequest, AccessControlResponse, ExtendedServicesRequest, ExtendedServicesResponse, Close.* The only valid extended services type for this interface is CIP ordering
Interfaces outside of ICS	The *OHS Translator* interfaces to the following ICS related elements: *Order Handling System*

3.5.5 UPS Translator Services

The following services are provided by the *UPS Translator*.

- Local user profile to CIP user profile
- Secure handling of authentication information
- Use of the local *catalogue* communication protocol for user information

The *Translator* contains elements which are common across ICS sites, e.g., those parts which interface to the *Retrieval Manager*, and contains parts which are unique to the site, e.g., those parts which speak to the local *UPS*.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with a *UPS*.

The *UPS Translator* has the functional interfaces indicated in Table 3-6.

Table 3-6 UPS Translator Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	An *UPS Translator* is capable of being a CIP *target* for one *CIP session*. Typically the *origin* will be a Retrieval Manager. This interface will be a later release of CIP..
Interfaces outside of ICS	The *UPS Translator* interfaces to the following ICS related element: *User Profile System*

3.5.6 HTTP/CIP Gateway Services

The following services are provided by the *HTTP/CIP Gateway*:

- CIP Messaging. The *HTTP/CIP Gateway* converts the user's inputs received via HTTP to CIP messages for transmission to a *Retrieval Manager*.
- HTML Generation. The *HTTP/CIP Gateway* dynamically generates HTML output from a set of templates for presentation of CIP information to the user. This may be based on CGI scripts (programs) capable of receiving input from - and sending HTML output to - any standard WWW browser.

- Session management. The *HTTP/CIP Gateway* maintains CIP session context in response to http messages which is a state-less protocol, e.g., preservation of information from screen to screen for a particular user session (e.g., result set IDs, user preferences). The HTTP protocol does not maintain session information, it is a one time passing of information. Virtually every single mouse click to a browser results in the gateway generating a single page from scratch.
- Security. The *HTTP/CIP Gateway* supports user authentication in the ICS.
- Attribute Management. The gateway provides an attribute cache as a local cache for the attribute definitions provided by the *Retrieval Manager*
- *CIP Client* server. Some *HTTP/CIP Gateway*s may act as servers of portable client code. One example is a Java applet which would allow a Java enable web-browser to run some or most of the *CIP Client* layers at the user's computer.

The *HTTP/CIP Gateway* has the functional interfaces indicated in Table 3-7.

Table 3-7. HTTP/CIP Gateway Application Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	*HTTP/CIP Gateway* is capable of forming CIP sessions with at least one *Retrieval Manager* where the *HTTP/CIP Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Interfaces outside of ICS	The *HTTP/CIP Gateway* may have interfaces to the following ICS related elements: *a web server*

3.5.7 ICS Gateway

The following services are provided by the *ICS Gateway*.

- mapping of local attributes to CIP attributes (e.g. for *product*, *guide* and *collection*)
- mapping of local query elements to CIP query elements
- mapping of CIP result sets into local result sets
- use of the local *catalogue* communication protocol
- The *ICS Gateway* converts the user's inputs received via http to CIP messages for the *Retrieval Manager*.

The *ICS Gateway* contains elements which are common across ICS sites, e.g., those parts of the *ICS Gateway* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

The *ICS Gateway* has the functional interfaces indicated in Table 3-8.

Table 3-8. ICS Gateway Application Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	*ICS Gateway* is capable of forming *CIP sessions* with at least one *Retrieval Manager* where the *ICS Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Interfaces outside of ICS	The *ICS Gateway* may have interfaces to the following ICS related elements: *catalogue* and *Existing agency client.*

3.5.8 Retrieval Manager Administrator (RMA) Operations

The following operations are performed by the *RMA*:

- Collection Definition. The *RMA* creates the site *collections* in accordance with the ICS Collection Manual, including site *provider archive collections, provider theme collections,* links to *collections* held at other sites.
- Collection Maintenance. The *RMA* maintains the site *collections* in accordance with the ICS Collection Manual, including periodically checking the *collections* held in their *Retrieval Manager*s for consistency.
- *Retrieval Manager* Operations. The *RMA* monitors and corrects any incorrect operations of the *Retrieval Manager* at their site.

The *RMA* has the operational interfaces indicated in Table 3-9.

Table 3-9 Retrieval Manager Administrator Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	The *RMA* accesses the *Retrieval Manager* via an operator's interface. The operator interface provides a graphical user interface to allow the *RMA* to monitor, control, diagnose, and maintain the operations of the *Retrieval Manager*.
Collection Management Tools	The *RMA* uses a *CMT* operator interface to control ingest of data into the *CMT*, modification of the data, and insertion of CIP *Collections* into the *Collection Database*.
Catalogue Translator	The *RMA* maintains the data and monitors the operations of the *Catalogue Translator* at the site.
OHS Translator	The *RMA* maintains the data and monitors the operations of the *OHS Translator* at the site.
UPS Translator	The *RMA* maintains the data and monitors the operations of the *UPS Translator* at the site.
Interfaces outside of ICS	The *RMA* may have interfaces to the following ICS related elements: *OHS*, *UPS*, *Catalogue*, *Archive*. These interfaces are not discussed here as they are outside of ICS.

3.5.9 Collection Management Tools (CMT) Services

The following services are provided by the *CMT*:

- Conversion of files into CIP compatible format *Collections*
- Support maintenance of the *Collection Data Base*

The *CMT* has the functional interfaces indicated in Table 3-10.

Table 3-10 Collection Management Tools Interfaces

Other ICS or Related Element	Interface Description
*Retrieval Manager*s	The *CMT* interfaces to the *Collection Database and Explain Database* of the *Retrieval Manager* to establish and maintain the EO content of the *collections* held in the data bases.
RMA	The *CMT* provides an operator interface to the *RMA*. The operator interface allows the *RMA* to control ingest of data into the *CMT*, modification of the data, and insertion of CIP *Collections* into the *Retrieval Manager*.
Interfaces outside of ICS	The *CMT* will have an interface to accept data which is to be made into CIP *collections*, e.g. EOSDIS V0 DAAC data set descriptions, DIFs. This interfaces is not discussed here as it is outside of ICS.

3.5.10 Monitoring and Control Tools (MCT) Services

The following services are provided by the *MCT*:

- Monitoring of *Retrieval Manager* operations and error handling.
- Commanding the *Retrieval Manager* to change states based on SSM commands

The *MCT* has the functional interfaces indicated in Table 3-11.

Table 3-11. Monitoring and Control Tools Interfaces

Other ICS or Related Element	**Interface Description**
Retrieval Manager	The *MCT* interfaces to the Retrieval Manager to gather operation and error handling data from the Retrieval Manager and to control the configuration and state of Retrieval Manager processes. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.
Translators	The *MCT* may interface to the translators to gather status and command changes similar to the interface with the *Retrieval Manager*. This particular interface is TBS.
Interfaces outside of ICS	The *MCT* may have interfaces to the following ICS related element: SSM. These interfaces are not discussed here as they are outside of ICS. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.6 Identification of ICS Related Element Services

This section describes the services assumed to be provided by Elements related to ICS but not part of ICS. For these elements which are shown on the ICS Framework (Figure 3-9) but are not part of the ICS per se, the following identification of services are assumptions which will be used as the ICS development proceeds.

- *Catalogue*
- *Order Handling System (OHS)*
- *User Profile System (UPS)*
- *Archive*
- *Site System Management (SSM)*

3.6.1 Existing Catalogue Services

The following services are assumed to be provided by the *Catalogue*:

- Directory service provides descriptions of data sets containing high level information suitable for making an initial determination of the potential usefulness of a data set for some applications. This information will be used to populate ICS *collections.*
- Inventory service provides the information about the data held in an archive. This information will be used as ICS *product* descriptor data.
- Browse service provides access to *browse* data.
- Guide service provides both *guide* descriptors as well as the *guide* data.

Note that a particular existing catalogue may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Catalogue*.

3.6.2 Order Handling System (OHS) Services

The following services are assumed to be provided by an existing *OHS*:

- Persistent Storage of Orders originating from a *Retrieval Manager*
- Formulation of quotes on orders
- Statusing of orders
- Billing and accounting of users orders (for orders filled by the *OHS* and by orders for which the local *Retrieval Manager* acted as a proxy.)
- Notification to users of order filled
- Invoices to users

Note that a particular *OHS* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *OHS*.

3.6.3 User Profile System (UPS) Services

The following services are assumed to be provided by an existing *UPS*:

- Persistent Storage of user profile data

Note that a particular *UPS* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *UPS*.

For a *UPS* that is based on a public-key infrastructure (PKI), the following services are assumed to be provided by a Certification Authority: (CA)

- Publish the criteria for granting user certificates
- Granting, revoking and general certificate management functions
- storing root keys

3.6.4 Archive Services

The following services are assumed to be provided by an *Archive*:

- Persistent Storage of EO data
- Filling of orders: delivery via ftp or media

Note that a particular *Archive* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Archive*.

3.6.5 Site System Management (SSM)

The following services are assumed to be provided by an existing *SSM*:

- Management oversight of functional elements at the data provider site
- Monitoring of site services
- Commanding of site service configuration and modes

Note that a particular *SSM* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *SSM*.

4. DATA VIEW

The Data View characterizes the data required to support the ICS functions as described in the Functional View Section of this document. The main data object in ICS is the *Collection.* The first section of this chapter provides a motivation from an EO user's point of view for *Collections*. This section also describes the Collections Model and then provides insight into how the users will interact with the model. This section is primarily concerned with the conceptual nature of the *Collectio*n. The next sections primarily address the logical nature of the *Collection* by providing a description of the data components of ICS. This is achieved by introducing a data object model for the relative data components, and by identifying the content and source of the content for each data object.

4.1 Collection Concept

This section provides the conceptual view of the EO Data structure by describing the motivation for the *collection* concept, by constructing and describing the collection model and lastly by providing examples of the collection concept using the EO Data. Collectively this information will provide the conceptual framework from which the details of collections can be discussed and the logical data discussion can begin.

4.1.1 Motivation for Collections

Users of Earth Observation data will be interested in subsets of data from various datasets and various locations. A user is typically interested in an assembly of data which best fits the topics which the user needs to investigate. This assembly requires the data to be organized in a certain way. It is this notion - that the users need for data does not uniformly overlap the organization - that provides the motivation for the collection concept.

The following are some examples of how EO Data consumers will want to access EO data:

- It is assumed that in general the user community will often desire access to specific groupings of ICS holdings. For example the user may request information from a single instrument, or for a specific wind-speed and direction data from AMSR, or of trace gas distribution data from MIPAS;
- Because agencies have responsibility for holding the data from certain sources, the physical location of data will be assumed to be geographically distributed around the world. Collections can be formed which can be associated with products from multiple geographic locations;
- It is assumed that the community will also require access which spans instruments. For example, a request for all sea surface temperature data, across MODIS, AMSR, ASAR, and so on;
- It is assumed that inter-disciplinary work will call for access across science disciplines and instruments, relative to some specific time, space, or event coincident coverage.

Therefore, to address these somewhat divergent but related needs the concept of a *collection* was developed. This concept embraces the following fundamental concepts.

- The *collection* is a container
- The *collection* has members

- The members can be other *collections*

These concepts allow the data to be organized in a manner which will facilitate the access needs of the user.

The following sections explore these concepts in further detail by introducing a model of the collections concept, describing the collection membership, and providing examples of user interactions with the collections.

4.1.2 Collections Model Overview

The collections concept permits a wide degree of flexibility in EO data organization. A *collection* is described in ICS as a collection descriptor, i.e., a set of attributes which describe the *collection*. The *collection* may have members which are reflected as included item descriptor identifiers. These identifiers serve to identify the associated *products*, *guides*, and *collections*, through the *Included Product Descriptors, Included Guide Descriptors*, or *Included Collection Descriptors* respectively.

A *collection* supports the notion of associated data or membership either due to their physical location (i.e. database storage), or along the lines of subject themes, that are deemed useful organizational principles. The science data provider coupled with a *Retrieval Manager Administrator* should decide the appropriate organizational strategy. Certain rules will be specified in the ICS Collection Manual to provide the necessary guidance to assist this team in making the appropriate decision.

Some examples for arranging data follows. Traditionally EO data was arranged according to EO product types (Raw, Level 0, etc.). Within ICS *Collections*, data can be arranged according to agency, discipline, spectral range, instrument, processing level, geographical area, etc. The user can then direct a search in a more focused manner, by including or excluding particular collections.

As a *collection* can contain either included *product descriptor* ID’s, included *guide descriptor* ID’s, or *included collection descriptor* ID’s, it is possible to associate a number of collections under a single theme as the data provider or user finds applicable. It is also possible to associate an existing set of *product descriptor* ID’s, which are already associated to an existing *collection*, to a new *collection* by just referring to the existing *collection descriptor* ID within the definition of the new *collection descriptor*. Because Release B of CIP does not support mixed collections each *collection* may only include descriptors of a specific type; i.e. *Product*, or *Guide*, or *Collection.*

Figure 4-1 illustrates this concept followed by Figure 4-2 which further illustrates the notion of a *collection* and its’ relationship (association) with other *collections.*

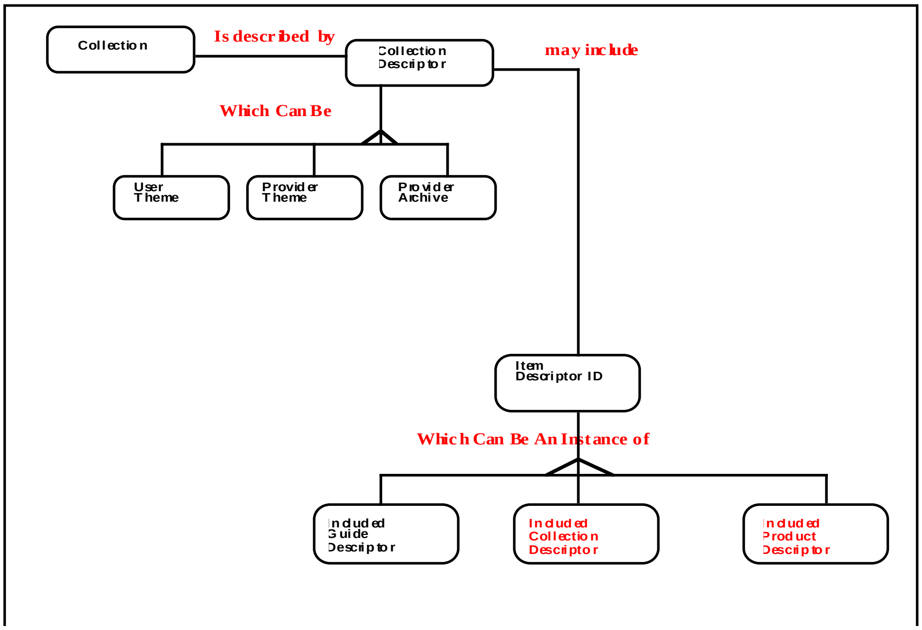


Figure 4-1. Single Instance of a Collection

The above model is an attempt to illustrate the concept of a *collection* which is illustrated in Figure 4-2. First the *collection* is described by a *collection* descriptor which logically consists of a group of attributes which describe the overall nature of the collection, such as, name, temporal and spatial extents. Further the descriptor can be classified as a *user theme, provider theme* or *provider archive*; definitions for these classifications can be found in section 4.2.1. These characteristics; definition and classification provide the overall description for the collection and the collections members. Next, the membership is identified by associating (including) *guide descriptors, collection descriptors, or product descriptors* of existing *guide, collections,* or *products*. Lastly, an included *collection* descriptor may be classified as a *user theme, provider theme,* or *provider archive.* The remaining sections will explore each of the above elements of the collection.

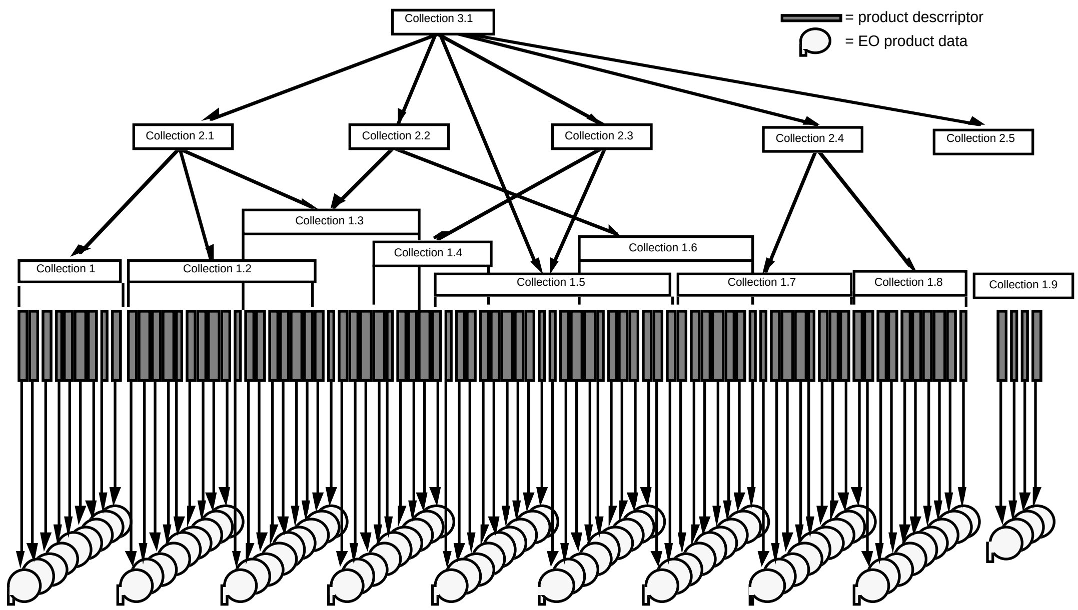


Figure 4-2 The 'Collection' Concept

Figure 4-2 expresses the logical relationship between multiple *collections.* As can be seen a web of interesting relationships can be formulated when using the collection concept. The *collections* in Figure 4-2 do not represent the naming of *collections* in an actual implementation. As can be seen the *collections* can overlap each other and can be associated with more than one collection. The central concept is that there are terminal collections, i.e., Collection 1.2 and non-terminal, i.e., Collection 2.2. The terminal collections describe the information contained in the inventories and are defined as Archive Collections in ICS. The Archive Collection Included Item Descriptor Id's will reflect only the Included Product Descriptor ID's. Above the terminal level collections, there are non-terminal collections. These collections are typically constructed around a theme, or event, however not limited to these aggregations, which expresses the primary tenet of the associated collections. In ICS these are the Theme Collections. The included collections may be any existing collection.

There is no restriction on the number of included item descriptors within collection tree levels, therefore a non-terminal collection could group together terminal collections and other non-terminal collections. Also, a terminal collection could exist without a relationship to a higher collection (i.e. collection 1.9), or a non-terminal collection could exist with no relationship to lower collections, in other words a collection without members (i.e. collection 2.5). These conditions could occur in particular circumstances, for instance when a catalogue site has only one single collection, i.e. Collection 1.9, or when the collection is under construction, i.e., Collection 2.5. The specific details for constructing and maintaining the collections are described in the ICS Collections Manual

4.1.3 Example Of Using Collections With EO Data

Several examples of constructing collections based on the users needs are reflected in Figure 4-3. In this illustration a Collection has been established to include MODIS data from several missions. Also a Collection has been created based on the theme Sea Surface Temperature (SST), which associates existing SST collections from AMSR, ASAR and MODIS. The intent of the subset caption is to show that only some the item descriptors are included. Lastly, a Collection is defined based on Andes Event. As illustrated, this collection includes collections which otherwise may not be related.

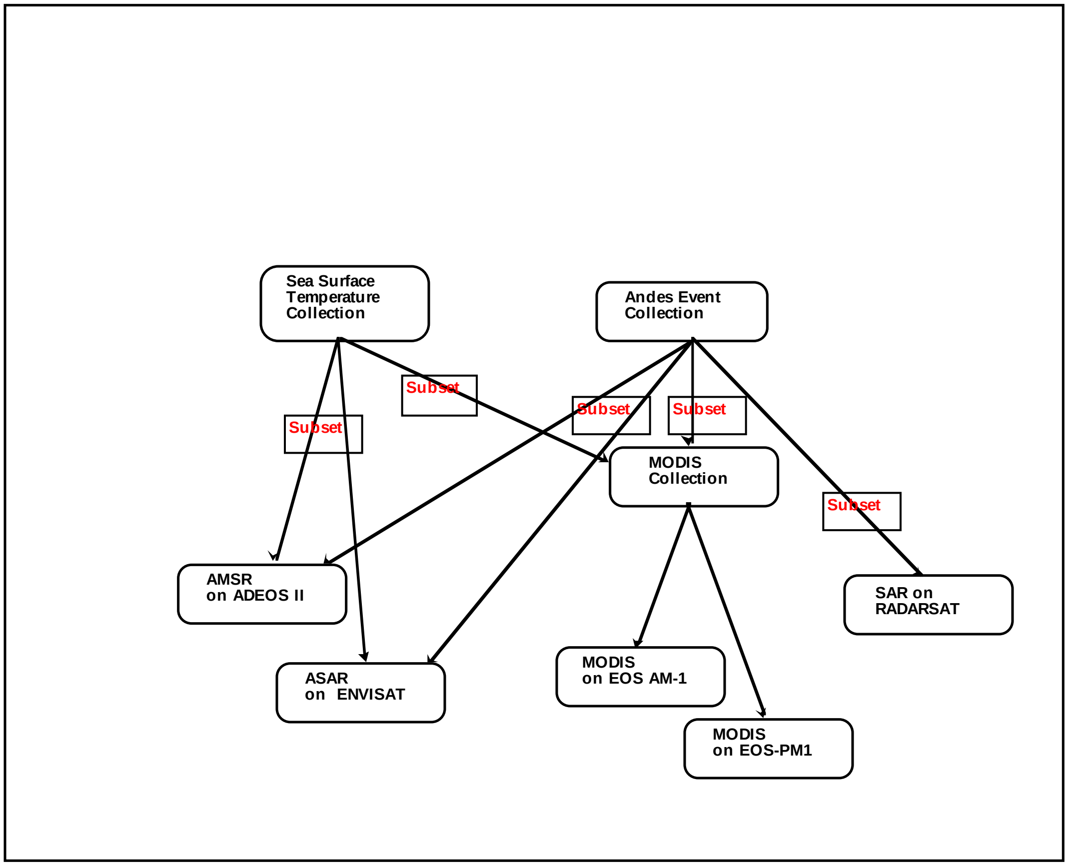


Figure 4-3. Creating Collections for Users

4.2 Collection Concept Details

In order to fully understand the power of the Collection Concept within ICS the following detailed information is provided in this section. The intent is to describe the collection descriptor classifications (categories), to provide some insight into collection membership, to discuss and define characteristics which are unique to this concept, and lastly, to describe how a user can interact with collections.

4.2.1 Collection Descriptor Classification

The flexibility of the collections concept requires an accurate classification scheme to allow the users of ICS to determine the characteristics of certain collections they are creating or viewing. In support of this requirement two major classification elements are used to classify *collections* in the ICS. These classification elements are owner and type of *collection*. The following describes each of these elements.

- Owner
    - *Provider*. A *provider collection* is owned by the *Retrieval Manager* site and is guaranteed to meet ICS Quality Assurance guidelines in metadata population and peer science review.
    - *User*. A *user collection* is owned by a user of a *Retrieval Manager*. These collections are highly heterogeneous both in metadata quality and science quality
- Type
    - *Archive* - An *archive collection* reflects a single underlying physical archive that is "owned" by a single *Retrieval Manager*
    - *Theme* - A *theme collection* reflects a common concept or theme and may include members from many underlying physical archives that are "owned" by several different *Retrieval Managers*.

It should be noted that the element classification combination of owner = "*User*" and type = "*Archive*" is not valid because the underlying *collections* are by definition owned by the provider. This leaves three valid combinations of elements; *provider archive*, *provider theme* and *user theme* as the valid classifications of ICS Collections. These classifications are further discussed in the remainder of this section.

Examples of collections which can be used to group data together which have a similar semantic theme can be seen in Figure 4-4 . The classification of those *collections* include:

1. *Collections* of SENSOR data by mission (e.g. MODIS from AM-1, SAR from RADARSAT, etc.) which reflects the organization of the underlying physical archives. *Provider archive* would be the correct element classification.
2. A *collection* has been established to include MODIS data from several missions. A collection has been created based on the theme of Sea Surface Temperature (SST) including SST products from AMSR, ASAR and MODIS. This collection would be classified as a *provider theme* collection.
3. A *collection* is defined based on an event. A new collection of otherwise unrelated products may be defined, based on the detection of a intense precipitation event over the Bolivian Andes. *User theme* would be the appropriate classification.

The first type of *collection* is likely to be created by data providers to organize their archives and facilitate access to the *product* descriptors (i.e. analogous to an inventory containing inventory entries). The second type of *collection* may be established by data providers who want to organize some of their data into groupings which differ from their *provider archive collections* (i.e. from the baseline inventory), for the convenience of their users. The third type of *collection* is likely to be created by an end user of EO products that has created the *collection* to obtain a single source of thematic information. This will then enable further analysis or ready access by themselves or other users. The following describes these collection classifications in more detail.

Provider archive collections. This type of collection is likely to be created by data providers to organize their archives and facilitate access to the *product descriptors* (i.e. analogous to an inventory containing inventory entries). A *provider archive collection* could be a dynamic collection where new item descriptors are automatically added to the collection by the catalogue system as new archive entries are created, i.e. for data from an existing, operational EO satellite. Alternatively, a *provider archive collection* could be a static fixed collection of historical data that is no longer being added to as the source satellite is no longer operational. A *provider archive collection* includes only item descriptors that are local.

Provider theme collections. This type of collection may be set up by data providers who want to organize some of their data into groupings which differ from their *provider archive collections* (i.e. from the baseline inventory), for the convenience of their users. For example, this *collection* could be based on the geographical area covered, the scientific discipline supported by the data, the instrument type, etc. The difference between *provider theme* and *provider archive collections* is that *archive collections* only contain homogeneous item descriptors, whilst *theme collections* may have item descriptors of differing formats and attributes. It is likely that there may be a large number that are homogeneous but not the complete *collection*. *Provider theme collections* can be terminal or non terminal dependent on how the data provider decides to organize their data or documents.

User theme collections. The third type of *collection* is likely to be created by an end user of EO products that has created and populated the collection to obtain a single source of thematic information. This will then enable further analysis or easy access by themselves or other users. These are collections of potentially quite disparate item descriptors of interest to a relatively small user community researching a particular theme, i.e. in the example, the mid-west flood of 1993. There may be item descriptors from different image archives, in situ measurement archives and bibliographic archives of relevant papers. The members of a *user theme collection* will in general be formed from the results of a series of searches to build up a set of descriptors.

Note that these classification definitions are not mandatory for the CIP to operate, however to ensure data interoperability the *collections* described above must share identical semantical interpretation across ICS. The CIP does not distinguish between the classifications(the *Retrieval Manager* does, however) and the same CIP search and retrieval services are applicable to all three. The *Retrieval Manager* does make use of *collections* for routing of distributed queries. Although a *collection descriptor* element is defined to hold a collection classification, the primary use of the categorization is to aid the ICS user in understanding the quality guidelines imposed on the *collection* data and metadata. Standardization of *collection* definitions is provided as part of the ICS design.

The three classifications of collections are illustrated in Figure 4-4.

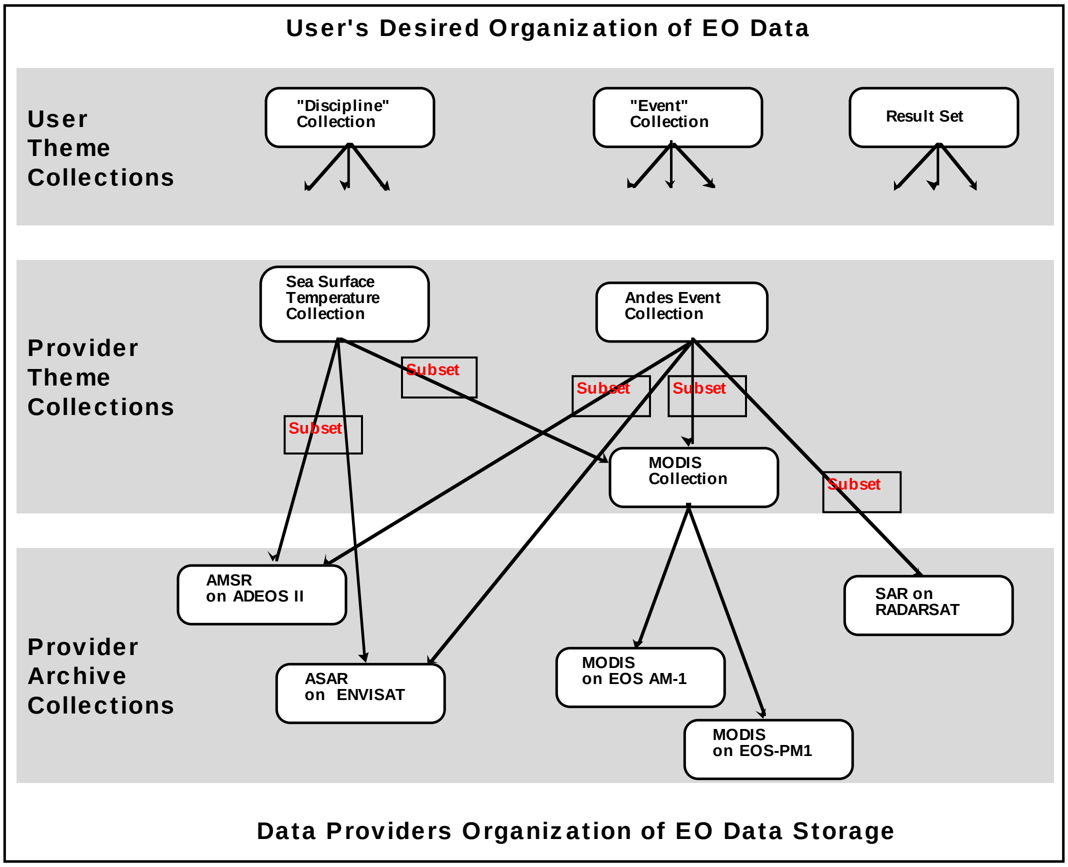


Figure 4-4: User and Provider Collections

4.2.2 Collection Members

Collection Membership in ICS is defined as including remote or local item descriptor ID's in the *collection descriptor* which defines the *collection*. A detailed process for identifying and including members will be described in the ICS Collections Manual. However, there are several fundamental rules which serve to guide this process.

1. A *product identifier* must be contained in one and only one *provider archive collection*.
2. *Archive collections* must include a reference to their associated *products*
3. *Theme collections* may include other *theme* or *archive collections* or *products.*
4. Mixed *collections* are not allowed in the Release B. (See Section 4.1.2 for an description of mixed collections)

All *product/guide* descriptors must be contained in at least one *provider archive collection* and therefore all archive collections must have a corresponding attribute definition table. The definitions of the attributes need only be stored once as they apply to all the homogeneous members of the archive collection and as they are stored in the *Explain database* they can be accessed by the *Retrieval Manager* as required.

In particular, *product/guide* descriptors must originate from a *provider archive collection* otherwise they cannot be accessed by a *Retrieval Manager*. This is because the definition of the attributes for any member are associated with the *provider archive collection* and not the individual member.

In particular, product/guide descriptors must originate from a provider archive collection otherwise they cannot be accessed by a Retrieval Manager. This is because the definition of the attributes for any member are associated with the provider archive collection and not the individual member.

Therefore, for non archive collections (i.e. provider or user theme collections), the product descriptor identifier must include an origin collection identifier which is the originating provider archive collection. For provider archive collections however, the originating collection identifier clearly does not need to be physically stored with each product descriptor, but only once as part of the collection descriptor.

For members of collections that are remote *collections*, item descriptor attributes (for remote *collections* and *product* descriptors of those remote *collections*) are logically held in the *Explain database* owned by the *Retrieval Manager* of the remote *collection*

4.2.3 Collection Characteristics

This section explores in more detail, the higher level concepts that were introduced in the previous sections. This is achieved by identifying and defining detailed *collection* characteristics to include the following:

- Commonality.
- Evolution over time
- Individuality.
- Member type.
- Collection trees.
- Identifier.
- Uniqueness.
- Remote members.
- Related collections.

Commonality:

By definition, a *collection* is a grouping of items that have something in common. A *collection* may have members that have many or fully common attributes (*provider archive collection* or *provider theme collection*), or a *collection* may have members that have a common semantic theme, though only a small subset of common attributes (*user theme collection*).

Further CIP specifies a list of standard attributes that can be searched. Some of these standard attributes are mandatory for all *collection* members (different mandatory sets for different descriptor types), while some are optional (although commonly understood). Finally, some attributes can be locally defined.

Evolution over time:

The members within a collection may be dynamic or static which will cause the nature of the *collection* to vary. The following describes each of these states.

Static members do not change over time. This can be a result of static underlying *collections* or the mechanism used to create the *theme collection* such as a Volcanic Eruption Theme Collection whose members will more than likely remain static over time.

Dynamic members may change over time based on changes in the included *collections*. It is envisioned that the majority of *collections* will not be static, but will evolve as ICS is used. This will occur in response to the way in which the user community wants to view relationships between the various data held by ICS, and the ways the CEOS Agencies wish to respond to those desires. Dynamic membership will require close supervision by the *Retrieval Manager Administrator* to ensure that the *collection* descriptor information is current.

Individuality:

A *collection* member(included item descriptor ID) may be a member of more than one *collection*, but duplicate members(included item descriptor ID's) are not supported within a single *collection*.

Collection trees:

As *collections* can contain pointers to other collections, reflected in the included item descriptor ID's, there exists the concept of a 'collection tree' (see Figure 4-2), the leaves of the branches being product or guide descriptors. The *collections* that include only *product* or *guide* descriptors are termed 'terminal collections'. The *collections* which include only other *collections* (i.e. no *product*/*guide* descriptors) are termed 'non-terminal' *collections*.

Identifier (In the following text, *item descriptor* covers *product, guide* or *collection descriptor):*

Each member of a *collection* (i.e. included item descriptor of any type) must have an identifier unique within all the *collections* in the *retrieval manager's* collection tree. This unique identifier can be seen as the name of the item descriptor. In particular, it will be a single unique identifier which includes the *Retrieval Manager* identifier, the *collection* identifier and the *collection* member identifier(included item descriptor ID's). For a *collection* descriptor, the *collection* identifier refers to a collection of any classification (*provider archive, provider theme* or *user theme*), whereas for *product* descriptors, the identifier must refer to a *provider archive collection*. Specific format details can be found in the CIP Specification[R3].

Uniqueness:

By virtue of the unique identifier, every *collection* existing can be uniquely identified in the domain of all *collections* (relevant for multiple-site operations). This is particularly important when considering the case of the *collection* which is a child of two or more *collections.* Any operation which traverses the *collection* tree will end up repeatedly visiting the child *collection*. The unique identifier provides a means of preventing repeated operations on the same *collection* by noting which tree nodes had been visited and then restricting access to those nodes for the same search. This reduction in revisiting will be performed by the *Retrieval Manager.*

Remote members:

Normally, a *collection* tree would be held in one place (say, as a database on a computer). A logical *collection* tree is where one or more of the members are held elsewhere - the complete *collection* tree thus spans multiple sites. The *collection* that is referencing(included item descriptor) a *collection* at the remote location is termed a 'remote member'.

Remote members do not have to maintain information about which members refer to them; remote members are indistinguishable from local *collection* members from the user's point of view. This concept is supported by the consistent use of URLs to identify *collections*, in the same manner as the complete WWW is seen by the user as a single database. A *Retrieval Manager* 'owns' those *collections* that it stores the attributes and values for locally, and only stores the pointers(Included Item Descriptor ID's) to remote members, not their attributes and values.

Within the complete CIP domain, attributes of *collections*, *products* and *guide* data should only be stored once, and controlled by the *Retrieval Manager* that owns the items. The only exception to this may be with the members of *user theme collections*.

No attribute or value of a remote member, or the pointer (included item descriptor) to the remote member, can be considered as guaranteed. The *Retrieval Manager* where the remote member is stored may not be available; the remote member may have changed its data structure (adding, changing or deleting attributes), or the remote member may have been deleted from the remote *Retrieval Manager*.

Related collections:

Collections may be related to one another without the need for a "parent-child" or the "include" construct. The relation may be through content or purpose for example and allows the spanning of one *collection* tree to another. A *collection* descriptor will contain a list (possibly empty) of related *collections* as part of its content.

4.2.4 User Interactions with Collections

This section describes multiple methods a user will have available using the CIP within the ICS domain to discover items in *collections*, e.g. *products*, *guide*, *browse*, etc. Four methods for using the *collection* structure are described:

- Collection Discovery
- Collection Navigation
- Collection Searching.
- Collection Locating.

Each of these methods differ based on what the user needs to know before using the method and what element of the *collection* descriptor is used by the method. An example is that for *collection* discovery the user needs to know very little of the ICS, compared with collection locating in which the user must have a specific *collection* ID to use the method.

These methods are depicted in Figure 4-5 which illustrates each of the *collection* usage methods identified above relative to a *collection* node. Each of the *collection* usage methods are described in the remainder of this section.

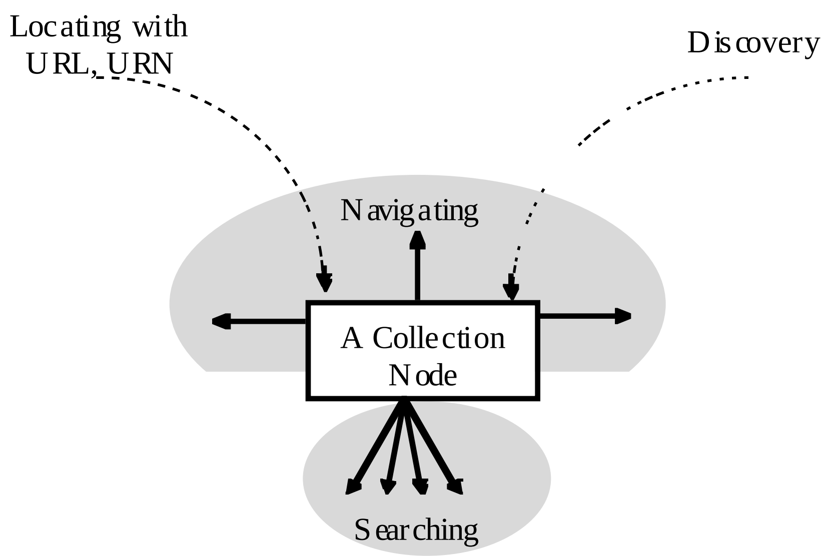


Figure 4-5: Collection Usage Methods

Collection Discovery:

Collection discovery allows a user of the ICS, having no prior knowledge of the *collection* structure, be assured that all *collections* of interest to the user are found. It is assumed that the user will know of at least one *Retrieval Manager*. This method provides a user with an exhaustive method to determine all *collections* which may be of interest. The discovery of *collections* makes no assumptions about the user knowledge of any existing *collections* or other *Retrieval Managers*.

For CIP-B, *collection* discovery will be accomplished by the relatively simple method of searching a global *collection* which is discussed in Section 4.10. It is anticipated that information discovery methods currently being researched in WWW community may have value for accomplishing *collection* discovery in later releases.

Collection Navigation:

Collection navigation allows the user to examine *collections* related to a specified *collection* and switch context to the related *collection*. For a given *collection* of interest, the user may retrieve a list of related *collections*. Navigation is similar to surfing WWW links from page to page where using the CIP a user can surf from *collection* to *collection*.. The related item descriptor is the data design construct which supports this functionality.

Collection Searching:

Collection searching is the core function of the CIP. With this method the user knows of a *collection* of interest as well as the *Retrieval Manager* where the *collection* descriptor for the *collection* resides. The user establishes a session with the *Retrieval Manager* and targets a search at the *collection* The search is then propagated down through the included item descriptors of the *collection* and any included *collections*.

Collection Locating:

Collection locating assumes that the user has a *collection* URL in their possession and wishes to find the *collection*, examine the *collection* elements, and perhaps target a search at the *collection*. Having a URL, the user can examine the URL and determine information about the *collections* location, the user can submit the URL to a *Retrieval Manager* which can locate the *collection* or determine that the *collection* no longer exists.

4.3 ICS Data Framework

This section of the Data View will present the logical data components of the ICS. This will be achieved by describing and defining each data component of the overall Data Framework. This framework as illustrated in Figure 4-6 represents the logical data objects which will be managed and maintained in ICS. The emphasis, in the illustration, and this section, is on the data held in the *Retrieval Manager* which includes the

- *Collections Database (CDB)* (Section 4.3),
- *Explain Database* (Section 4.4),
- *Extended Services Database* (Section 4.5),
- Session, Error and User Data (Section 4.7).

The data required by elements other than the *Retrieval Manager*, external interfaces for example, are recognized in Section 4.7. Each data component will be described in the sections indicated above. These descriptions will include an overview of the contents of the database, an object model, which will illustrate the data objects, attributes and relationships for each of these data repositories, and lastly any unique design characteristics surrounding these databases.

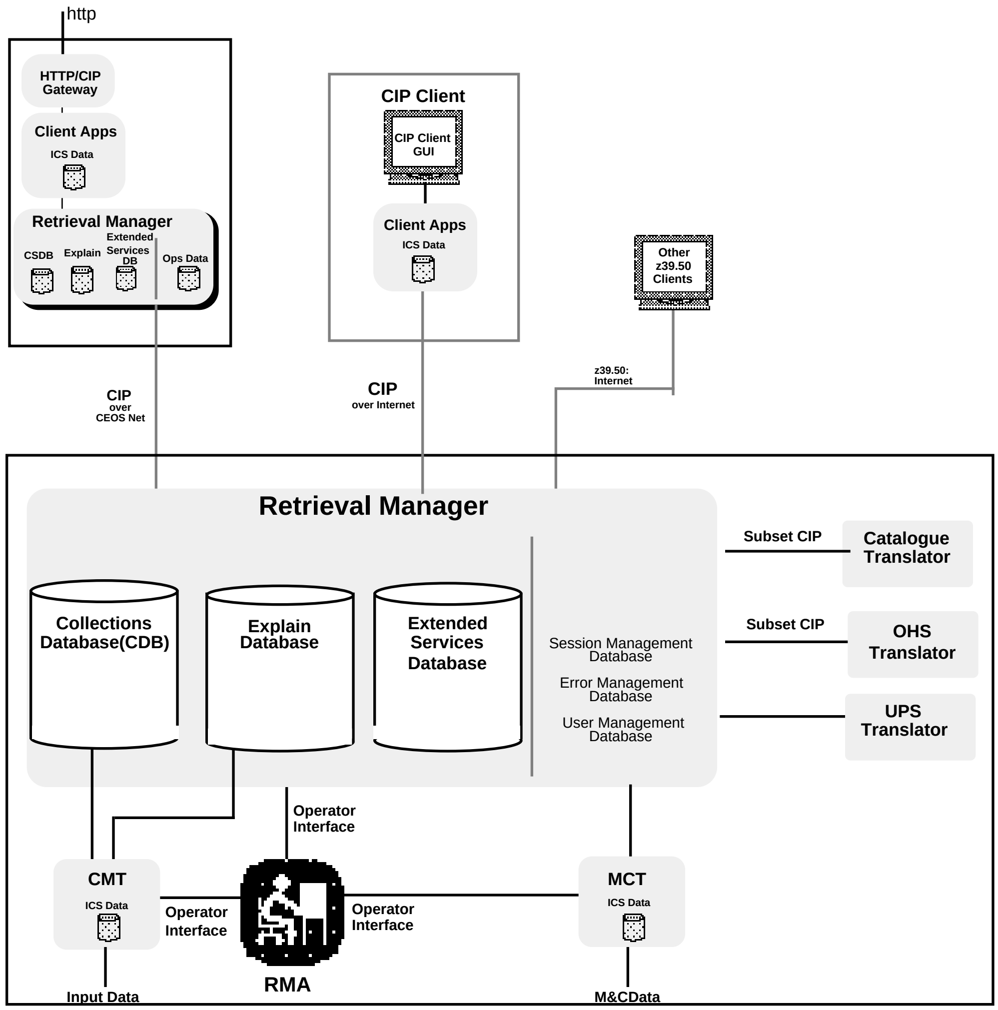


Figure 4-6. ICS Data Framework

4.4 Collection Database (CDB)

The *collection database* contains the instances of the science metadata which are referenced from the model illustrated in Figure 4-1. The metadata coupled with the organization of the metadata within the *collections database* is to assist the user in discovering, searching, and ordering Earth Observation information. The intent of the *collections database* is to satisfy these objectives within a framework which will allow maximum flexibility while minimizing change to the overall structure. This is achieved through the item descriptor which is a generalization of the specific classes of data which are supported under the *collection database* umbrella. The item descriptors supported within CIP's *Collection Database* are the *collection*, *product*, and *guide* Descriptors. The following definitions provide the context for each of these descriptors.

Collection Descriptors	Collection metadata serves to characterize the underlying product information/included collections. These characteristics consist of spatial, temporal, source, quality, document references, and keyword informational elements.
Product Descriptors	Product metadata characterizes the individual products contained in the archives. Each product can be described in terms of it's spatial, temporal, source and quality attributes.
Guide Descriptors	Guide metadata describes the keywords and review/revision cycle for the Guide documents.

For any of the above descriptors, a standard set of attributes are required (see CIP Specification - Release B [R3]). Note that not all catalogue systems will support all attributes, but the minimum set of attributes as specified in the CIP Specification should be supported. In the case of *product* descriptors (the attributes of which are based upon currently existing inventory entry definitions), there may be attributes that are local to a particular catalogue system. The CIP will handle these in a consistent manner, i.e. permit the transfer and searching of these attributes without needing to 'understand' their real world meanings[1].

An example of how the attributes can be used is presented in the tables below. (Note that the attributes represented in these examples are for illustrative purposes only. These examples do not include the full list of mandatory attributes. The complete result schema is defined in the CIP Specification Release B, Appendix C [R3], which indicates those attributes that are mandatory or optional.

The list of *product* descriptor attributes that might describe an AVHRR data product, and the corresponding values for a single instance of that product, might be:

Product Descriptor Attribute	**Example Value**
SensorName	AVHRR
StartDate	1994-07-10
StartTime	13:51:45
MissionId	NOAA-09
SpatialCoverage	-32.27, -79.39, 62.55, 65.32
ArchivingCentreId	OBERPFAFFENHOFEN
ItemDescriptorId	cip://ciprm.noaa.gov/PID.103.47c6dec1

Similarly, a list of the guide descriptor attributes and their values that might describe a guide entry explaining how the related AVHRR instrument works, could be:

[1] The CIP must be capable of integration within existing systems using the traditional 'inventory' approach, in which metadata describes products (i.e. where the two types of data are distinct). This is also appropriate for the 'search' for metadata and the 'order' of products. However, there are no constraints on the underlying catalogue or database structure and the CIP is designed with flexibility in mind. For instance support of a catalogue where the metadata and product are synonymous, i.e. the product descriptor contains the product, is a CIP Release B requirement although the CIP Release A design will facilitate this future extension.

Guide Descriptor Attribute	Example Value
Author	Dr. J. Smith
Title	NOAA AVHRR Data User's Guide
Abstract	This NOAA AVHRR Data Guide was prepared by the Canada Centre for Remote Sensing, 588 Booth Street, Ottawa, Ontario, K1A 0Y7. It provides the following information for the NOAA AVHRR sensor: - satellite/sensor points of coverage; - details on the acquisition of data and processing steps; - technical information on calibration details, accuracy and product format; - ordering information and user assistance; - contacts for other related products available internationally. All inquiries, references to omissions or additions should be referred to the address above.
PublicationDate	1994-07-10
VersionId	1.0
GeneralKeyword	AVHRR, CLOUD, IMAGE, GLOBAL, LAND, NOAA, RADIATION, VEGETATION INDEX
DeliveryFormat	HTML
ItemDescriptorId	cip://ciprm.noaa.gov/PID.103.47c6dec1
DocumentPointer	http://www.ccrs.nrcan.gc.ca/gcnet/guides/avhrr/avhrr.html

Finally, the EO collection descriptor for a *provider theme collection* may be part of a *collection* that contains all AVHRR data covering the European continent, a list of the *collection* descriptor attributes and their values that might describe such a collection could be:

Collection Descriptor Attribute	Example Value
ItemDescriptorId	cip://ciprm.esrin.esa.it/121
ItemDescriptorName	AVHRR EUROPE
CreationDate	1994-05-01
CollectionHierarchy Position	NON-TERMINAL
CollectionCategory	Provider Theme Collection

Collection Descriptor Attribute	Example Value
Purpose	This collection is a grouping together of all known AVHRR data that is available via the ESRIN and DLR EO data centres.
RevisionDate	1995-01-15
VersionId	2.3
Originator	ESA/ESRIN/DPE/OD
GeneralKeywords	AVHRR, CLOUD, IMAGE, GLOBAL, LAND, NOAA, RADIATION, VEGETATION INDEX, EUROPE
SpatialCoverage	-10, -70, 35, 70
Progress	IN WORK
BrowseId	http://gds.esrin.esa.it/avhrr/collections/europe.jpg
GuideId	http://www.ccrs.nrcan.gc.ca/gcnet/guides/avhrr/avhrr.html
CollectionContents	cip://ciprm.esrin.esa.it/226 cip://ciprm.dfd.dlr.de/125

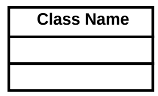
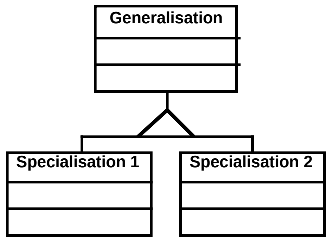
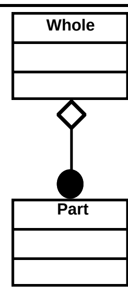
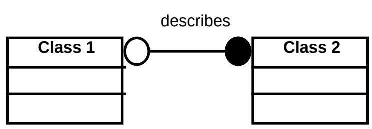
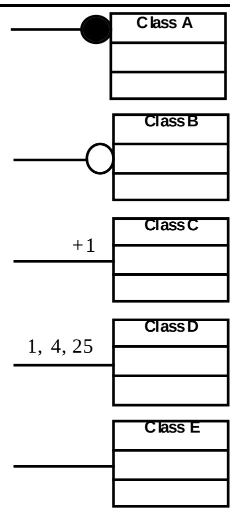
The list of attributes for each of the three item descriptor objects (*product* descriptor, *guide* descriptor and *collection* descriptor) will be different, as the information carried by the three objects is inherently different. It is also unrealistic to assume that (apart from any mandatory subset) the list of attributes for any one object within one agency will be exactly the same within another agency; this is especially true for the *product* descriptor objects, where existing archives and inventory entries are very different.

The CIP Specification - Release B specifies a set of attributes for describing *collections*. Not all servers need to support all *collection* attributes, however a mandatory set must be specified (see Appendix C of [R3]). For *product* descriptors there are a number of mandatory attributes that shall ensure basic interoperability between *product* descriptors (see Appendix C of [R3]). It is recognized that there are likely to be many *product* descriptor attributes that are local to a *catalogue* site or common to only a small number of *catalogue* sites (these will not be included within the CIP specification).

The Collection Database Data Object Model, which is illustrated in Figure 4-7.1 through 4-7.5, expresses the EO data objects, attributes, cardinality, and relationships between the objects. There are five modules which collectively represent the ICS CDB. The modules allow a partitioning of the entire CDB into manageable subsets. Description of these objects and attributes are recorded in Appendix “B” and “C” of the CIP Specification - Release B [R3]. The ARS’s which is the basis for these models is also contained in the Rel B Specification, Section C.3.[R3} The Business rules associated with the creation and maintenance of these objects are defined in the ICS Collections Manual.

The following table identifies and describes the Object Modeling Technique(OMT) notation that is used in the data models contained in section 4.4.1. Column one contains the OMT term, column two the graphical interpretation of the term and column three a brief description of the notation.

OMT Notation

Class-&-Object		Defines a class and its associated objects which provide an abstraction of something within the problem domain. The class contains attributes (data) and methods (functions) which are common between all objects in the class. Example: Collection is a class.
Generalization Specialization		Defines generalization - specialization relations between classes. The specialization classes inherits the attributes and services of the generalization class. A generalization class can have one or many specializations. A specialization can have one or more generalizations. Example: a Map Projection is a specialization of Planar System Class.
Whole-Part		Defines whole-part relations between classes. Example: Processing Options, Processing, and Scene Selection, are a part of Product Delivery Options.
Association Name		Identifies the relationships among classes... A relation has two labels, forward and inverse. For Example, Class 1 describes Class 2 is a forward direction,; the opposite would be Class 2 is described by Class 1 which is the inverse. These labels help to define the role of two or more classes with respect to the relation.
Multiplicity of Associations		Multiplicity specifies how many instances of one class may be related to a single instance of an associated class. For Example An instance of a collection may relate to one or more Spatial Coverage’s. The examples on the left express the ways in which multiplicities may be specified: Class A=many(zero or more) Class B= optional(zero or one) Class C=one or more Class D=numerically specified Class E=exactly one

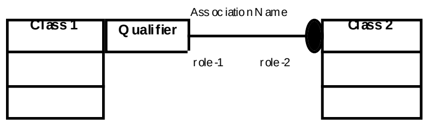
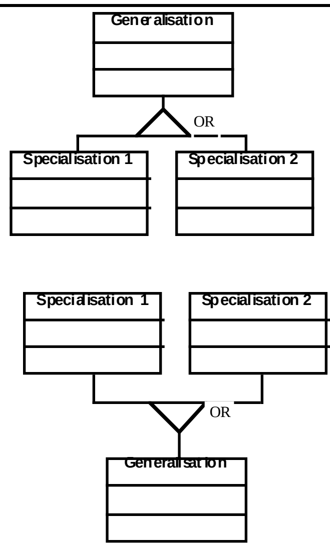
Qualified Association		Role names identify the participation of the objects in the overall association. For example The relationship between Guide and Contact states that each Guide must have at least one Contact who is an author. Qualifier is an attribute which serves to further qualify the association. For example the association between the Product class and Product Delivery Options class is qualified by the attribute Group ID
Exclusive Or		Exclusive "OR" indicates that only one of the specialization classes may exist for a given instance of the generalization class. This is graphically illustrated by indicating OR next to the specialization symbol.

Figure 4-7.1 reflects the *Collection* module which illustrates the data objects, attributes, and associations related to the *collection*. This model reflects the breadth of classes and relationships that could exist for any instance of a collection. The off page classes, Spatial Reference, Spatial Coverage, Database Info(Explain) and Item Descriptors(Extended Services) are illustrated in other *CDB* figures, *Explain Database* or *Extended Services Database* figures. The Item Data and Browse Pointer are references to instances of browse information that are characterized outside of the *CDB*.

Figure 4-7.2 reflects the *Product* module which illustrates the data objects, attributes, and associations related to the product. The same explanation provided above also applies to this module with the exception of the Database Info which does not relate to the product.

Figure 4-7.3 reflects the *Guide* module which illustrates the data objects, attributes, and associations related to the guide. The Contact class off page connector refers to the Contact class illustrated on the *Collections* Module.

Figure 4-7.4 and 4-7.5 illustrates the Spatial and Spatial Reference modules respectively and their individual data objects, attribute, and associations. In the Spatial Coverage Module the Spatial Coverage class is a special purpose class which exists to add clarity to the module. It will not contain attributes and therefore will not have instances other than the indicated specialization's. The G Polygon is also reflected in this module to add clarity and will not contain instances other than through the specified relationships Outer G-Ring and Exclusion G-Ring. The Spatial Reference Module contains several special purpose classes which are identical in purpose to the classes describe in the Spatial Coverage Module. The

Horizontal Coordinate System, Vertical Coordinate System, Spatial Reference and Planar System are all special purpose classes which serve to add clarity/definition to the module.

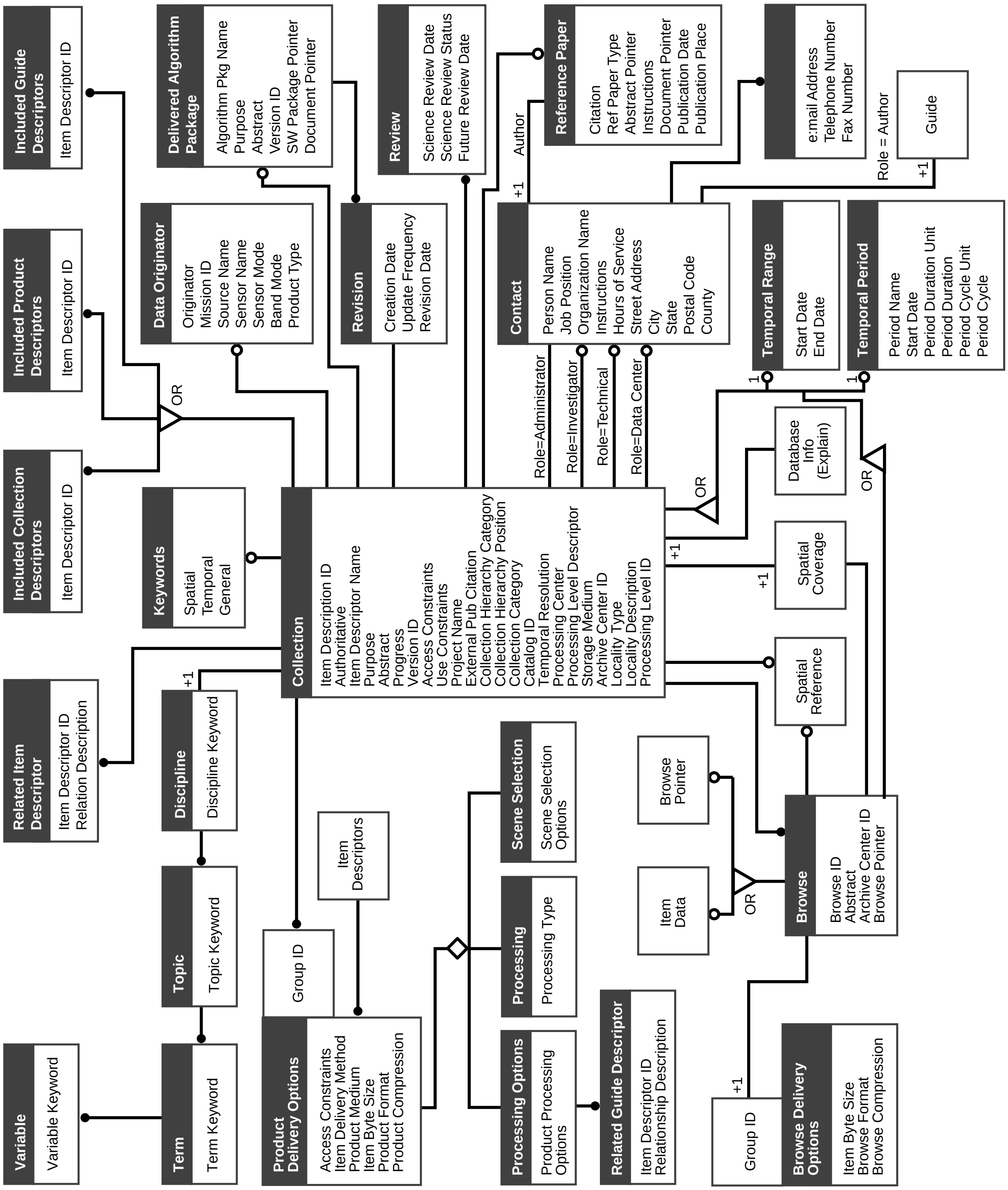


Figure 4-7.1 Collections Database - Data Object Model Collections Module

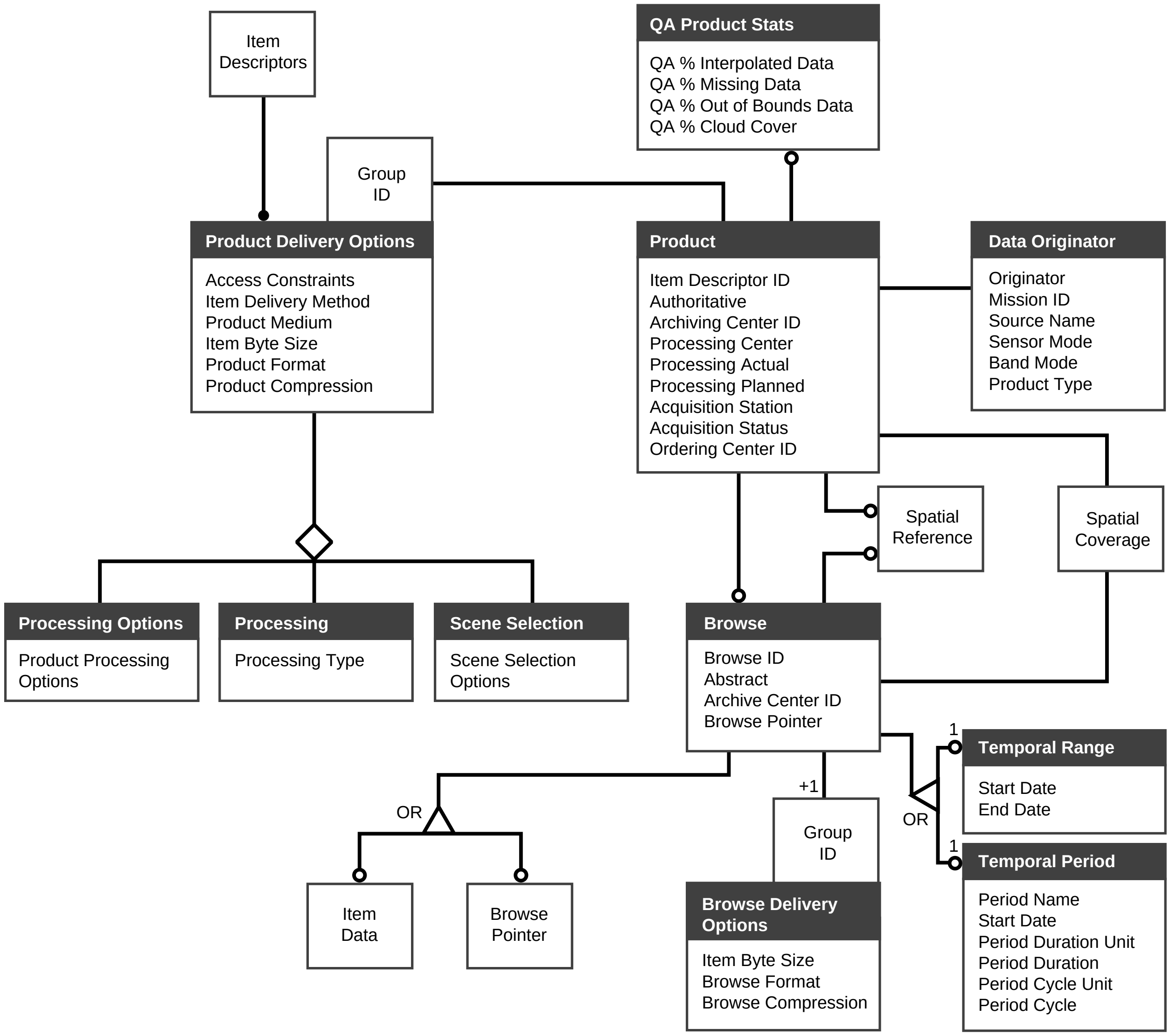


Figure 4-7.2 Collections Database Data Object Model - Product Module

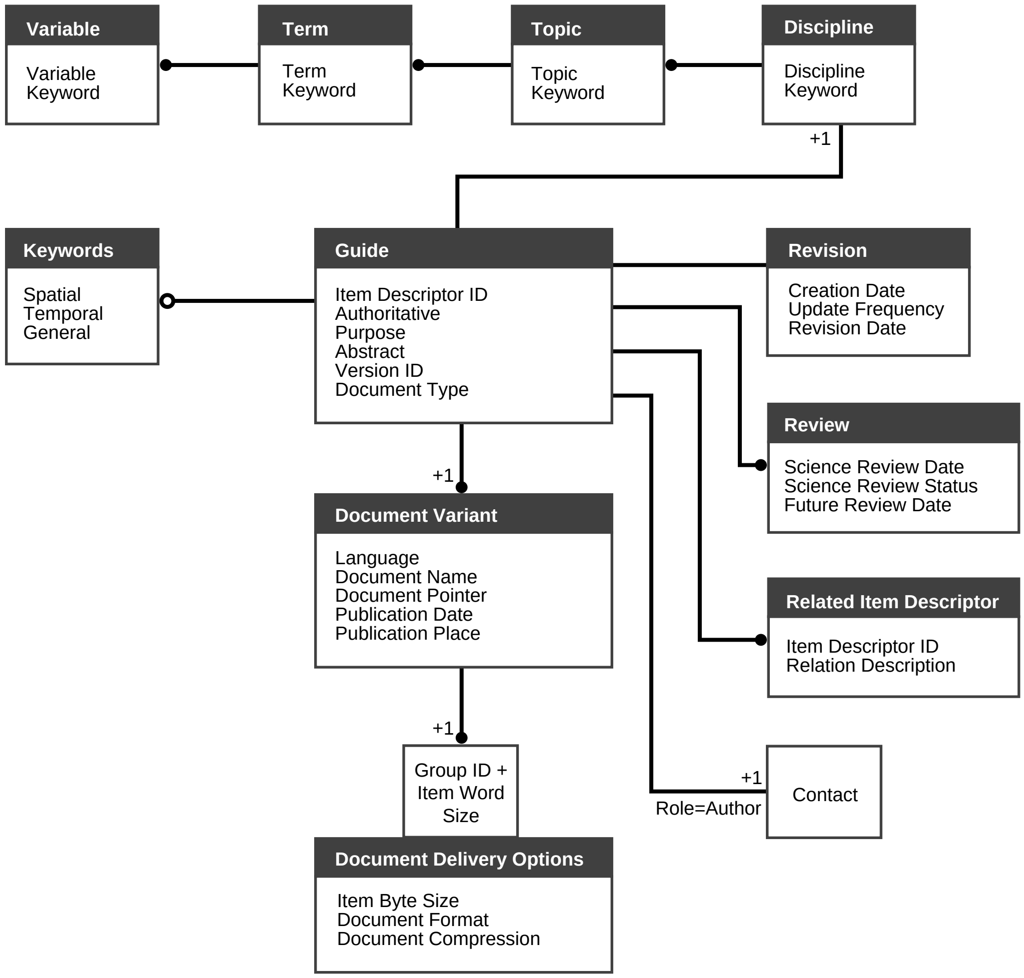


Figure 4-7.3 Collections Database Data Object Model - Guide Module

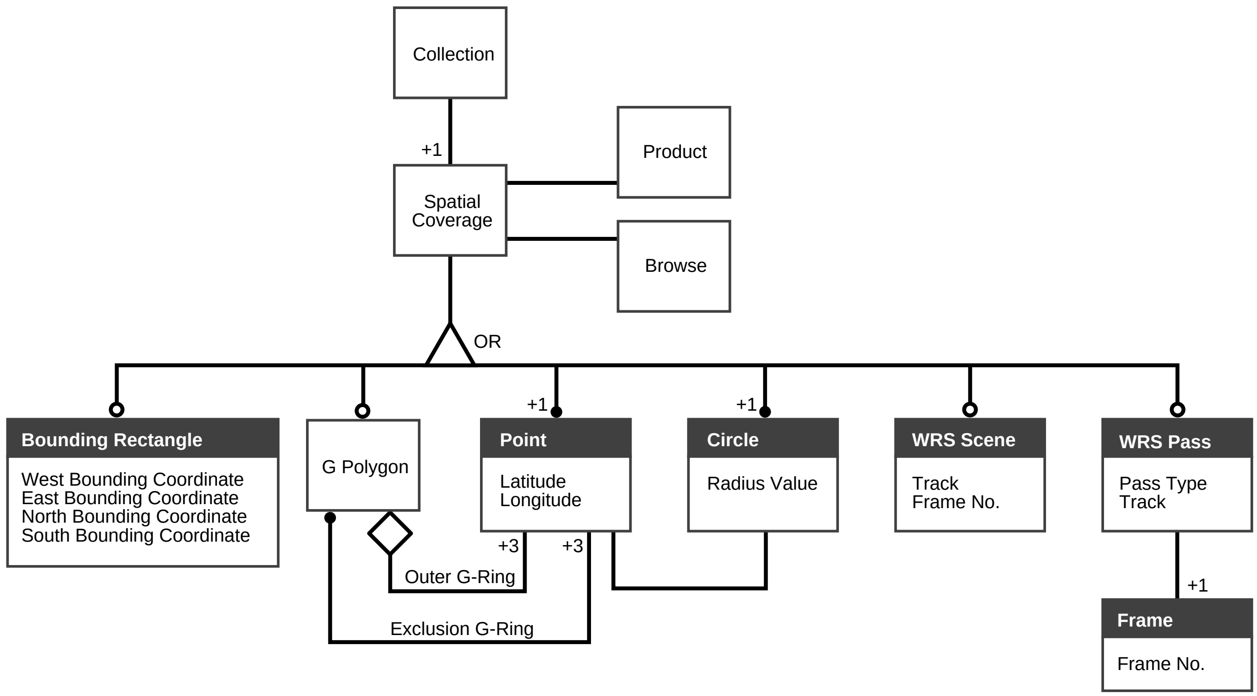


Figure 4-7.4 Collections Database Data Object Model -Spatial Coverage Module

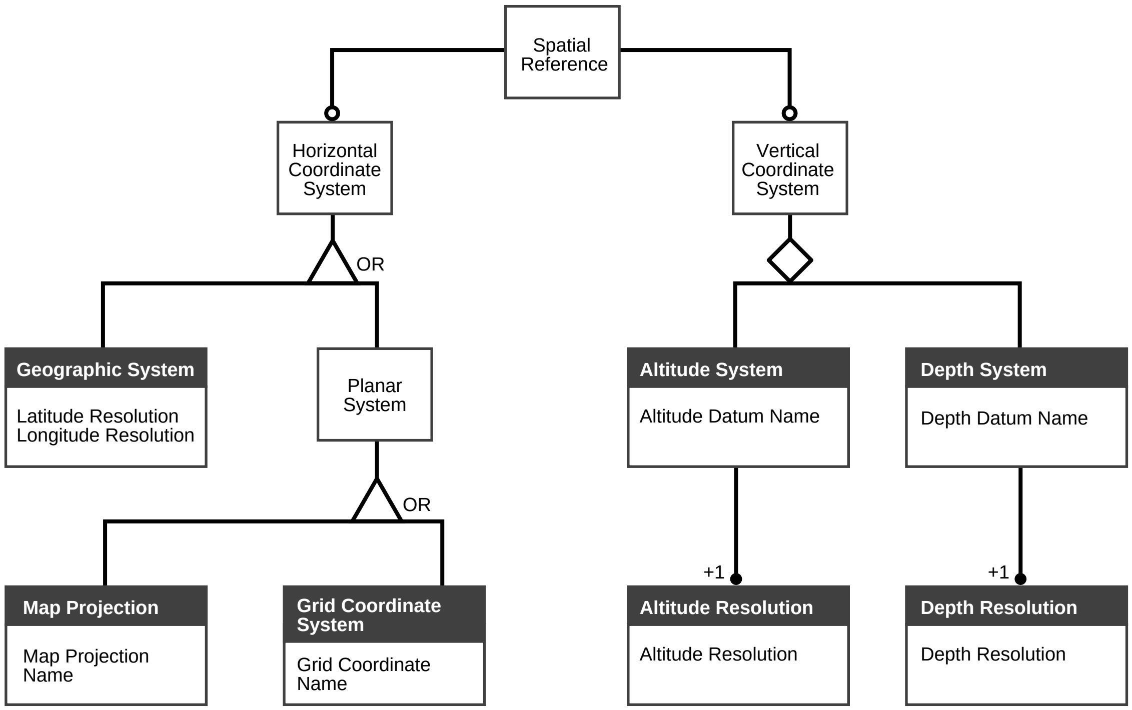


Figure 4-7.5 Collections Database Data Object Model - Spatial Reference Module

4.4.1 Collection Structure

Inherent in the *collections database* is the concept of a *collection* structure (Figure 4-8). A *collection* structure is defined as the logical grouping/organization of the item descriptors. The primary purpose of the structure is to organize the data in the CDB to support the search and present functions of the system. The structure represents the application of the data model within the ICS, which is presented in Figures 4-7.1 to 4-7.5.

As illustrated in Figure 4-8 each ICS Site will support a *collection* structure which consists of:

One root node which will contain the actual instances of the *Collection* Descriptor Data and

Guide Descriptor data, or all collections;

and, optionally instances of *Product* Descriptors (Product metadata).

This is the logical representation of the *collection* concept as represented in figure 4-2. The *collections* are the items contained in the root node, where as the *product* descriptors are in the product node. Actual instances of EO product data will reside in various archives. The inventory information(metadata) for those archives may also reside in various locations. The CEOS design approach is to avoid replication of the inventory requiring that queries will need to be distributed to multiple sites and that result sets are combined for presentation to the user. The mechanism for distributed query and result set processing will be transparent to the users. By defining *collections*(logical grouping of metadata and references to remote metadata) which can span sites, the *collection* concept, as described in Sections 4.1 and 4.2, provides a unified view of the data to the user while CIP and the *Retrieval Managers* will process the queries and result sets. Towards this goal the ICS supports the notion of Global Node, Root Nodes, Product Nodes, and Included Collections.

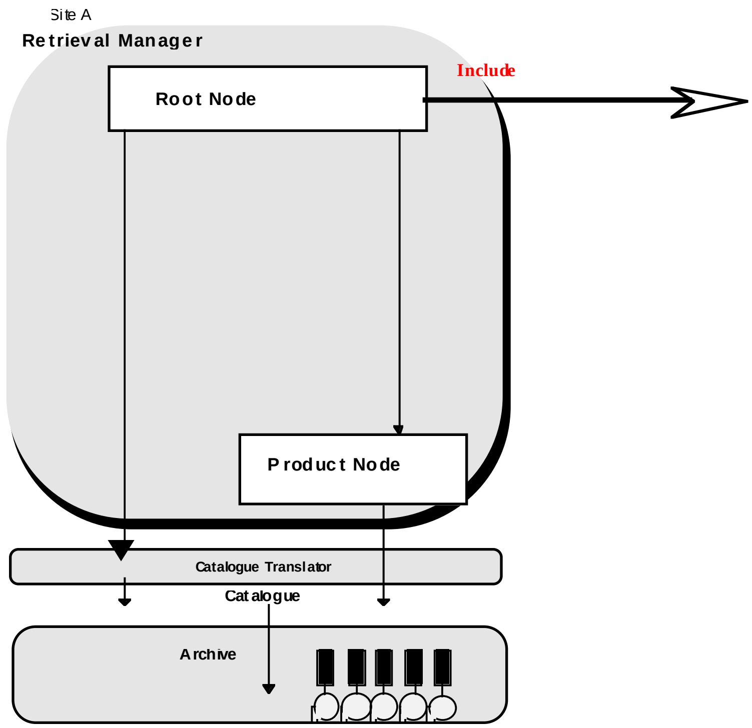


Figure 4.8. Site Collection Structure

4.4.1.1 Root Node

An integral part of the collection structure is the Root Node. The overall purpose of the root node is to allow a *Search Request*, which was formulated at the origin, to ultimately determine the EO *collection/guide* that match the criteria specified in the query elements of the *Search Request*. The root node serves as the logical mechanism which specifies the data contents through the item descriptor for all collections/guides that are owned by the *Retrieval Manager*. Therefore, the criteria specified in the query element are evaluated against the contents of the Root Node to determine if the *collection/guide* satisfies the *Search Request*. The data objects model for the root node contains the *collection* descriptor object and attributes, in addition to, the *collection* descriptor's associated data objects and attributes. Each EO *Collection* will be described in the root node as illustrated in Figures 4-7.1, 4-7.3, 4-7.4, 4-7.5.

Specific rules and the required procedures for developing and maintaining a root node are identified and discussed in the ICS Collections Manual. Descriptions for each object and attribute can be found in the CIP Specification - Release B, Appendices "B" and "C"[R3].

4.4.1.2 Product Nodes

The *product nodes* which contain the data objects, attributes and relationships which are illustrated in Figures 4-7.2, 4-7.4 and 4-7.5, identifies the *product* descriptor data objects and attributes that will be available to the *searchRequest* for *product* metadata requests. The *product* nodes are optional, however, if they exist then they must be related to a instance of an EO *collection* which is contained in the root node. Descriptions for each object and attribute can be found in the CIP Specification Release B, Appendices “B” and “C”[R3].

4.5 Explain Database

The *Explain Database* offers the ICS the structure and information necessary to respond to the Z39.50 Search, Segment and Present Services. This component of the ICS data architecture “Explains” the data or information environment in which the ICS will operate. This environment consists of categories of information that the target supports. These categories would include databases, schema’s, record syntax, attribute sets, and extended services. For each of these categories, a detailed description, which is contained in the attributes, relationships, and associated data objects serve to provide the model under which the ICS will capture, store, and make available information about the ICS *Retrieval Manager*. The *Explain database* data model which is illustrated in Figures 4-9.1 to 4-9.2 graphically depicts these concepts. Additional details regarding each of the objects in the data model are described in the CIP Specification - Release B [R3].

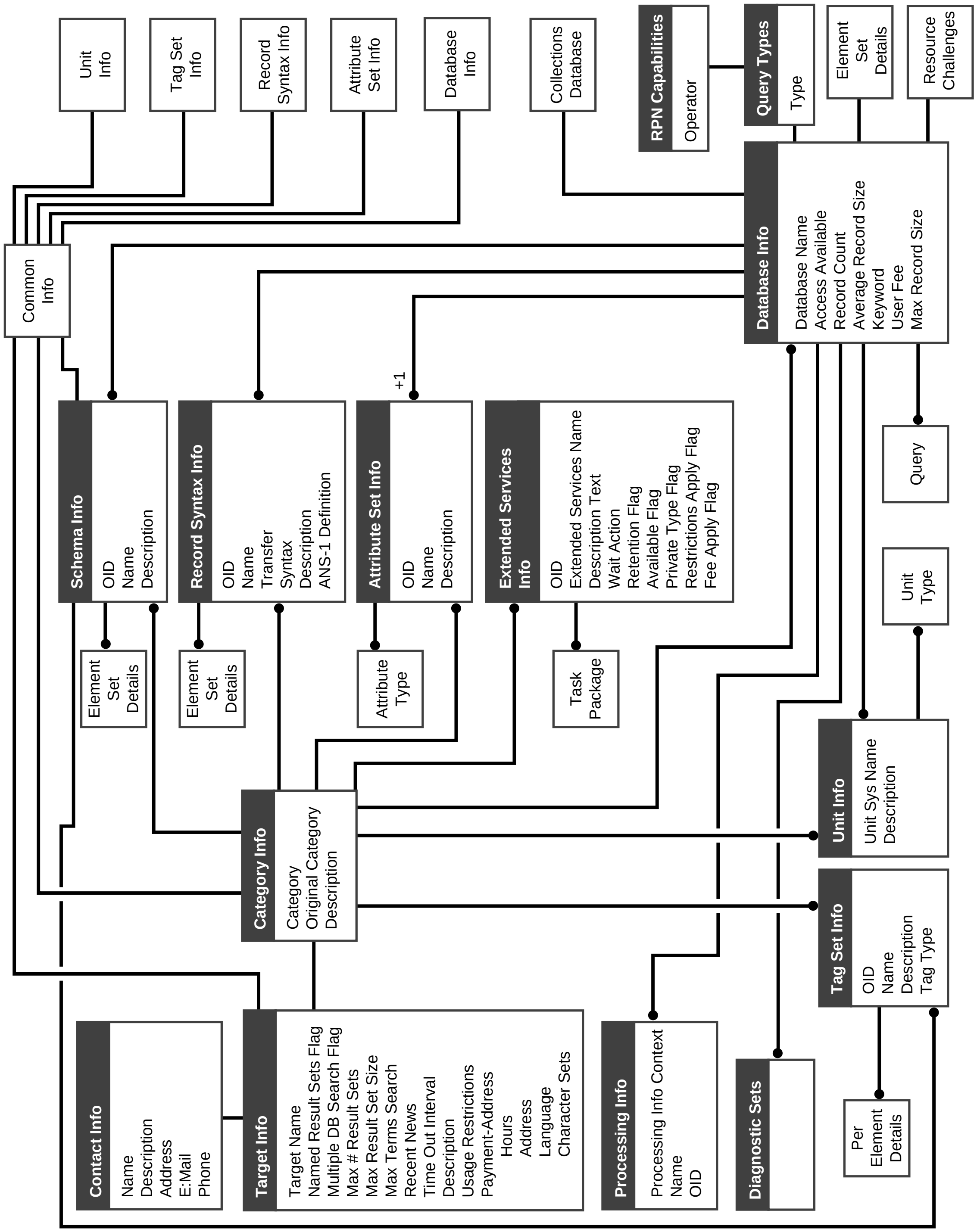


Figure 4-9.1. Explain Database

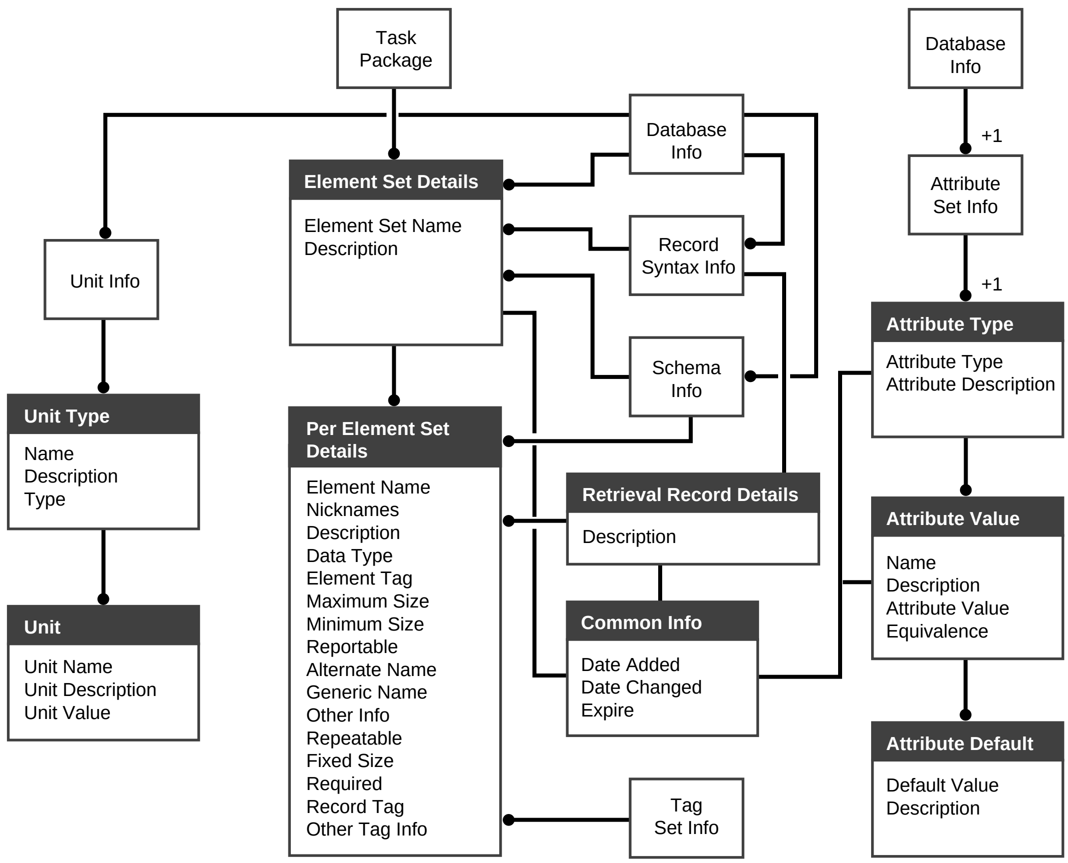


Figure 4-9.2 Explain Database

4.6 Extended Services Database

The *Extended Services Database* component of the ICS Data Architecture provides the information necessary to support the Z39.50 Extended Services Facility. This facility offers to ICS the extensions that are referred to as Task Packages, necessary to support the persistent storing and accessing of Order, Query, Database Update and Result Set information. The *Extended Services Database* contains a description of each of these task packages. The persistent result set task package contains the metadata about the instances of result set that are stored in the persistent Result Set Database. The Order Task Package contains the information necessary to respond to an order request from an origin. The Query Task Package provides the metadata associated with the persistent queries which are stored in a query file. Lastly, the Database Update Task Package provides a journal of the database activity associated with an ICS database. Each of these packages may have assigned a User or a Group of Users and their assigned privileges. Figure 4-12 graphically illustrates the *Extended Services Database* data object model. Additional details regarding each of the objects in the data model are described in the CIP Specification - Release B [R3].

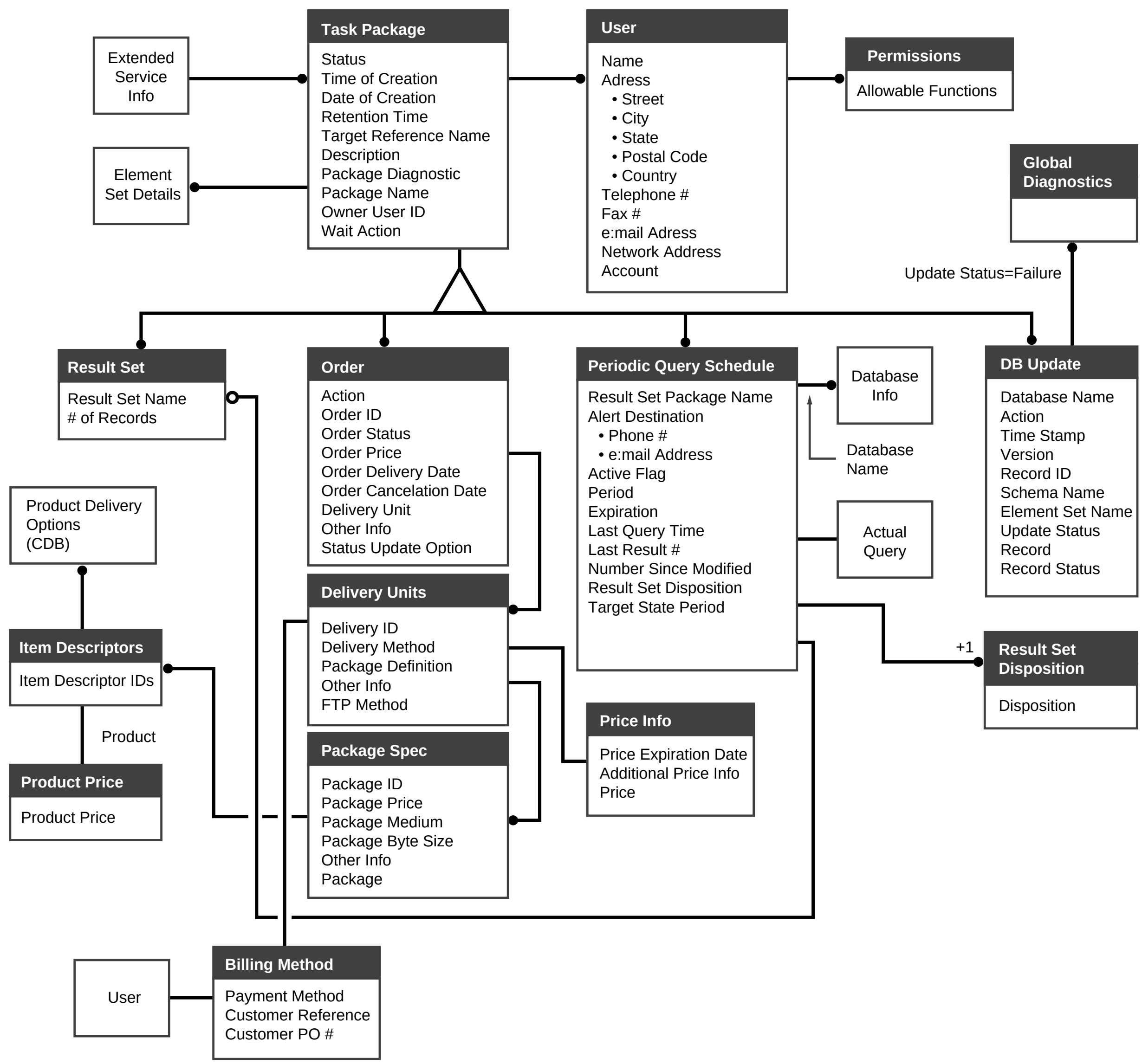


Figure 4-10. Extended Services Database

4.7 Session Management Database

The session management data will contain the information necessary to support Query, Results and General Session activity. The intent of this data are to provide the necessary information to support the interaction between the *Retrieval Manager* and the interfaces external to the *Retrieval Manager*.

- Query Data is characterized as information required to support the query processes to include both search and present. Additionally, Query Management information would also be included which would reflect Routing, and Status of Queries

- Results Data will include all of the information required to support results processing.
- General Session Information will include User Interaction Session logs which would contain the User ID, Location, Start and End Times, Profile of Tasks Performed, and Resources Used. Additionally, the CIP User Log which would capture Authentication Attempts, Successes, Failures, Service Options Accessed and Unsuccessful Log Ins would also be included, as well as command control information such as types of requests (order, batch, status query).

4.8 Error Management Database

The error management data are the data required to provide a description of error conditions which may occur during processing. This includes diagnostic messages, level of priority of error or failure and the location of the error within the system to include attribute error, software error, or hardware error. CIP Diagnostic Messages are listed in Appendix E of the CIP Specification - Release B [R3].

4.9 User Management Database

User Management data includes the User Profile information including the data required for authentication and user privileges. Additionally, *Retrieval Manager* profile data such as locations are also included in the user management database.

4.10 Global Node Data Architecture

A Global Node mechanism was chosen by the PTT to support Collection Discovery functionality within ICS. The intent of the Global Node is to answer fundamental questions about the location, and contents, of a *retrieval manager* or *managers*. The global node contains information which references the location of the *Retrieval Manager's* Root Node. It is assumed that the Global Node's *Retrieval Manager* will look much like any other *Retrieval Manager*. The major benefit of this approach is that it minimizes data replication and thus the risk of accessing out of date information. The Global Node in it's simplest form will serve as an information agent when a *Retrieval Manager* or the contents of a *Retrieval Manager* are not known. An illustration of this concept is contained in Figure 4-11.

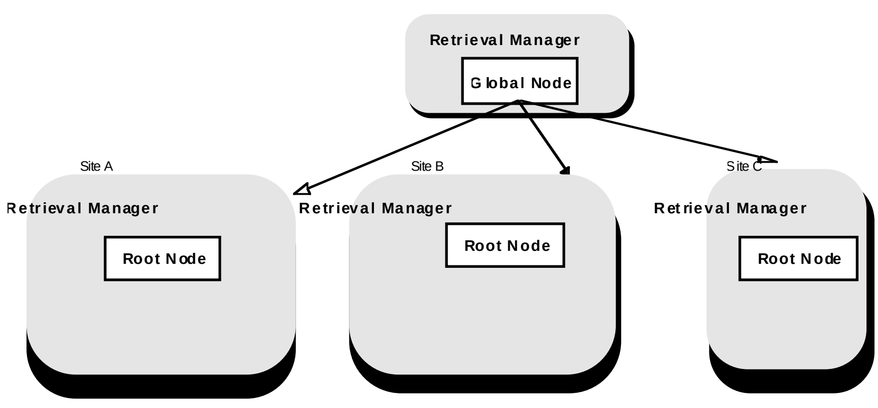


Figure 4-11 ICS Global Node

4.11 Collection Census

This section provides statistics on the number of collections anticipated in ICS along with parameters about the collection structure.

4.11.1 Number of Collections

It order to support the design analysis, a rough order of magnitude estimate of the number of collections in ICS is needed. A specific count of the collections is not necessary nor is a list of named collections. The estimates provided here are also not aimed at indicating an expectation that a particular agency will provide access to the number of collections indicated. The purpose of the estimate is to provide a high end estimate of what the CEOS collection space might grow to. This estimate is provided in Table 4-1.

The estimate reflected in Table 4-1 was discussed during the CEOS Catalogue Sub-Group meeting in May 1996. (The Catalogue Sub-group has been renamed the Access Sub-group.) It was the conclusion of the Catalogue Sub-Group that the estimate in Table 4-1 significantly under estimated the number of collections which could be indexed using CIP. In particular, the GCMD experience indicates that there are millions of datasets, although a large portion of these datasets will never be accessible on-line. The number of datasets which may provide on-line directory metadata may be on the order of hundreds of thousands. And for datasets which would have inventory on-line, an estimate on the order of tens of thousands was felt to be accurate.

For now the best that can be said is that CIP should be sized for hundreds of thousands of collections with collection level metadata only. Further, CIP should be sized to hold tens of thousands of Collections containing product descriptors.

Table 4-1. ICS Collection Upper Bound for Design Sizing

Agency	Rough Estimate of the Number of Provider Archive Collections
European	
BNSC	**100 +/- 50**
CEO	**500 to 5000**
CNES	
DLR	
ESA	**50**
Canadian	
CCRS	**500**
Japanese	
NASDA	**70**
US Agencies	
NASA (EOSDIS)	**Version 0: 300** **Version 1: 2000**
NOAA	**1200**
USGS	
Total	**Roughly 10,000**

4.11.2 Collection Structure Parameters

Also of interest is a projected structure of the total set of collections listed in the previous section. Some key parameters which characterize the structure of the ICS collection space is listed in Table 4-2.

Table 4-2. ICS Collection Structure Parameters

Collection Structure Parameter	Estimate of Parameter
Total Number of Retrieval Managers	Low: 10's based on one Retrieval Manager per agency High: 100's based on many Retrieval Managers containing one or two collections
Avg. Number of collections per collection	Low: 5 High: 50 (May be much higher for a Global collection)
Avg. Collection Depth from Global Collection (see equation below)	Low: 2 to 3 (assumes uniform tree with 30,000 total collections and 50 avg. collections in a collection) High: 6 to 7 (assumes uniform tree with 30,000 total collections and 5 avg. collections in a collection
Average Number of Product Descriptions per Provider Archive Collection	TBD

The following equation was used to calculate several parameters in Table 4-4. For a uniform tree, with D as the collection depth and A as the number of collections per collection, the total number of collection, T, is

$$T = \sum_{i=0}^{D} A^i \quad \text{(see Collection TN [R11] for details)}$$

Also of interest is the average number of products in a collection which will need to be estimated based on the type of collection under consideration. It is anticipated that an *archive provider collection* will have many more (orders of magnitude more?) members than a *user theme collection*.

4.11.3 Metadata Sizing

Also important to ICS element sizing is the size of product metadata, i.e. the size of a product descriptor. This number will certainly vary for the various products. For a first order estimate, an estimate from ECS is provided here. ECS is using an estimate of 2K/product. This estimate is being used for both ECS Release A (predominately Version 0 and TRMM data) and for the ECS DBMS prototype for ECS Release B (Release A plus EOS AM-1, Landsat-7, and others).

4.12 Data in Other ICS Elements

The *CMT, MCT, Retrieval Manager Administrator*, and *Client Application* elements of ICS will all require data to support the services identified in Section 3. The following briefly describes the data associated with each of these elements.

4.12.1 CMT

The data manipulated by the *Collection Management Tools* are contained in the *CDB*, *Explain Database, and Extended Services Database*. The tools will support the management of *collection* information contained in these repositories. This will include as a minimum, modifying *collection* entries, inserting new *collection* entries and deleting existing *collection* entries. Additionally, these tools will allow the *Retrieval Manager Administrator* to monitor the state of the *collections*. The details associated with the *Collection Management Tools* will be reflected in CMT Requirements and Design Documents. The SDD will be updated to reflect data design decisions if necessary.

4.12.2 MCT

The MCT will contain data which is specific to the suite of *Management and Control Tools* selected for ICS. The details associated with these tools will be specified in the ICS Administrators Manual. The SDD will be updated to reflect data design decisions if necessary.

4.12.3 Retrieval Manager Administrator

The *Retrieval Manager Administrator* will require interaction with the various components of the *Retrieval Manager*. The interaction with the Science Data Components will require that *CDB, Extended Services Database,* and the *Explain Database*, be made available to the *Retrieval Manager Administrator* to support data maintenance as well as general monitoring activities. The Operational Data Component interaction will involve the monitoring and controlling of the Session Management Data and Error Management data. The User management data will be available to the Retrieval Manager Administrator for management activities. The procedures and guidelines will be specified in the Administrator Manual. The data requirements will be specified in the Requirements and Design Documents for the RM. The SDD will be updated to reflect data design decisions if necessary.

4.12.4 CIP Client Application

The *CIP Client Application* will contain the necessary data to support the client functionality. For example the search service requires that the set of Use Attributes be made available to the Client Application. The data details will be specified in the Client Application Documents. The SDD will be updated to reflect data design decisions if necessary.

5. COMMUNICATIONS VIEW

The Communication View provides two models of the ICS: 1) a communication framework showing the protocol stack used in ICS, and 2) a description of the CEOS Network and Internet usage between *Retrieval Managers*.

5.1 ICS Communication Framework

This section defines the communications stack to be used for implementing CIP over the networks between ICS *Retrieval Managers*. The ICS will be implemented using the TCP/IP set of protocols. Note that the CIP is required to not be limited to TCP/IP (see URD UR 336 in [R2]), allowing some CIP applications to be implemented over an OSI communication stack for example. But, ICS shall exclusively be TCP/IP.

5.1.1 TCP/IP Services

This section provides an overview of the layering of CIP over TCP/IP. This will support the discussion in the next section on specific TCP calls by CIP.

TCP/IP has 4 conceptual layers of software. Starting at the top is the application layer, supported by the Transport layer, supported by the Internet Layer, and the Network Interface Layer. A hardware layer supports this stack. (See [R12] for a more extensive review of TCP/IP.)

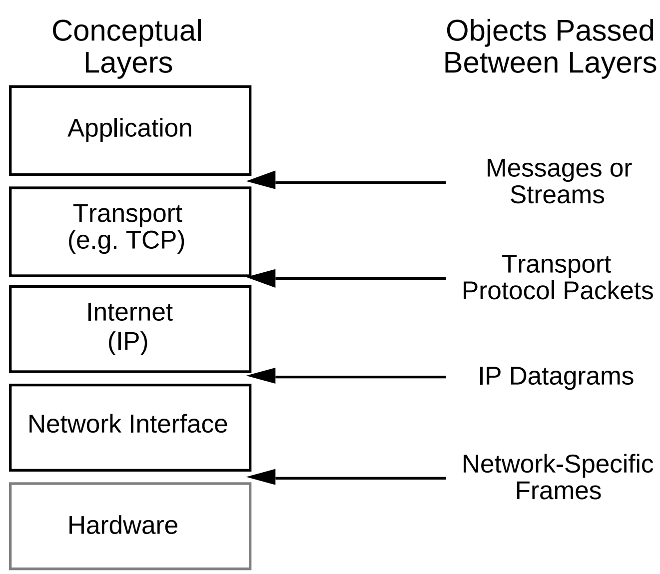


Figure 5-1. TCP/IP Internet Layering Model

5.1.2 CIP Translators and TCP Communication Stack

The implementation of the communication protocol stack for ICS is shown in Figure 5-2. CIP Applications within the *Retrieval Manager* or *CIP Client* will create CIP messages which can be encoded using Z39.50. Z39.50 is also considered an application level protocol and produces a byte stream which can be passed to the Transport Layer using a TCP socket. TCP establishes a virtual circuit with the remote site and sends Transport Protocol Packets to the Internet layer. The Internet layer implemented with IP, passes IP datagrams to the Network Interface.

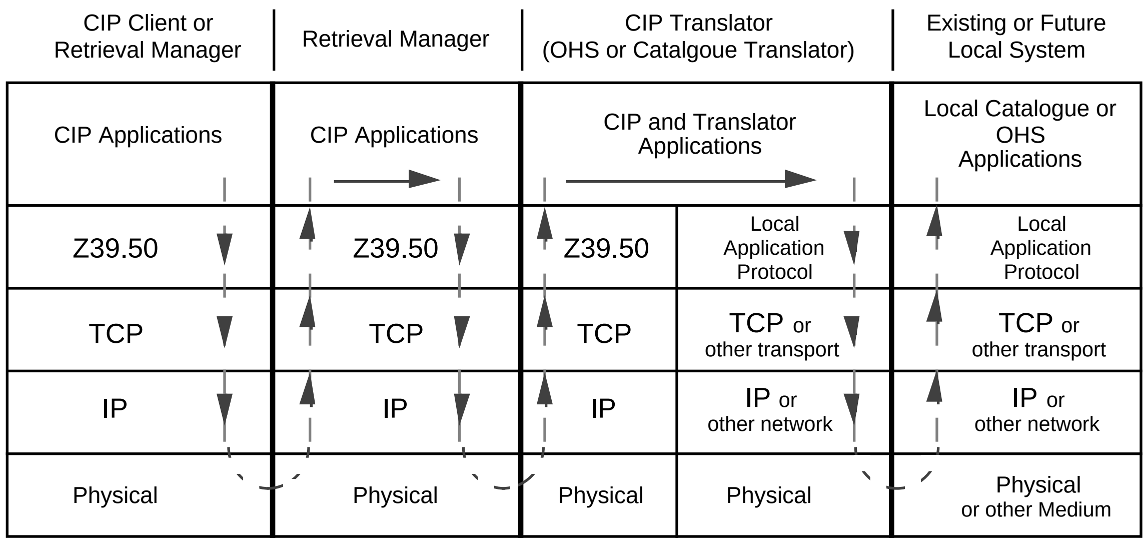


Figure 5-2. ICS Layered Communication Model

The specific example shown in Figure 5-2 shows a *CIP Client* (or a *Retrieval Manager*) establishing a *CIP Session* with a *Retrieval Manager*. Messages are passed down the protocol stack in the *CIP Client* with each layer encapsulating the message with its layer-specific information until they can be passed to the physical network to the other site. Note that there may be multiple devices in between the physical devices hosting the Client and the *Retrieval Manager*, e.g. routers. The messages are unpacked on the way up the protocol stack in the *Retrieval Manager*. In this particular example the message continues on from the *Retrieval Manager* to a CIP Protocol Gateway through a packing and unpacking cycle. In the application level of the CIP Protocol Gateway, the message may under go much translation to change the application semantics of the message from the CIP domain to the legacy system domain. The CIP Protocol Gateway then wraps the legacy application system message in the protocol specific to the legacy system communication stack. This allows the message to be passed form the Gateway to hosts within the local system, which in turn unwrap the message and answer the message at the application level.

5.1.3 Implementing CIP using TCP/IP Services

This section defines the interface between CIP and TCP services. This section summarizes a memo by Clifford A. Lynch [R13] and extends the topics to CIP. (Lynch's memo was written for Z39.50 Version 2 and has not been updated for Version 3.)

Encoding. The CIP specification and the Z39.50 standard specify application protocol data units (APDUs) in Abstract Syntax Notation One (ASN.1) [R14]. These APDUs include EXTERNAL references to other ASN.1 and non-ASN.1 objects such as those defining record transfer syntaxes to be used in a given application association. Standard Basic Encoding Rules (BER) [R15] are applied to the ASN.1 structures defined by the CIP profile and Z39.50 protocol to produce a byte stream that can be transmitted across a TCP/IP connection.

Connection. In the Internet environment, TCP Port 210 has been assigned to Z39.50 by the Internet Assigned Number Authority [R16]. To initiate a *CIP Session* with a *Retrieval Manager* in the TCP/IP environment, a *CIP Client* opens a TCP connection to port 210 on the *Retrieval Manager* and then, as soon as the TCP connection is established, transmits a *initializeRequest*. The TCP connection can be closed by either the *CIP Client* or the *Retrieval Manager* by sending a *close* message and then closing the TCP connection. Establishing a connection between two *Retrieval Managers* is handled differently and is addressed in the next session on distributed session management.

Data Transmission. All CIP messages, other than *initializeRequest* and *close*, involve the transmission of data over the established TCP connection. This is accomplished using TCP transmit and receive operations. The TCP facility for out-of-band data is not used.

5.1.4 Distributed Session Management

The need for Distributed Session Management in ICS is illustrated in Figure 5-3. A user may request a search which results in sub-searches to other *Retrieval Managers*. To accomplish this, the *CIP Client* forms a *CIP Session* with a *Retrieval Manager*. In order to send the sub-query to the second *Retrieval Manager*, a second *CIP Session* between the two *Retrieval Managers* is established. In the first *Retrieval Manager*, the second *CIP Session* must be associated with the initial *CIP Session* between the *CIP Client* and the *Retrieval Manager*, so that results are returned appropriately to the client which initially requested the search. The set of *CIP Sessions* needed to achieve the users request is referred to as a *User Session*.

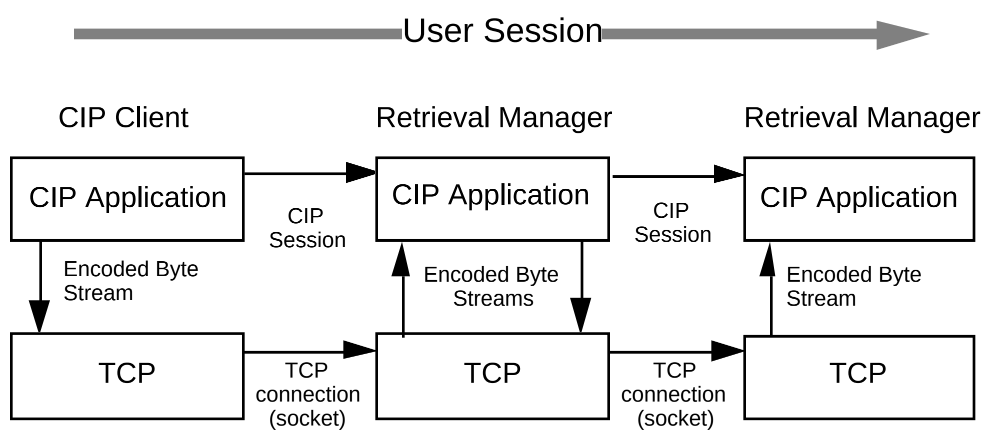


Figure 5-3. Distributed Sessions

The management of distributed sessions is accomplished in the following fashion.

- Individual *CIP Sessions* are based on Z39.50 associations. Z39.50 is a single client to single server protocol, i.e., it does not provide for distributed session management across several *Retrieval Managers*.
- CIP defines a unique, mandatory Reference ID for each message, which is used to manage message tracking.
- The *Retrieval Manager* is required to manage associations between incoming CIP messages and subsequent secondary messages to other ICS elements. This is accomplished using the session management logs defined in Section 4.

So, the management of distributed sessions is accomplished by the session logs which are established in the *Retrieval Managers*. A session log is established for a user during the initialization of a session. For each CIP request which the *Retrieval Manager* receives it maintains the association with any secondary request in the session log. When the secondary response is returned, the session log is consulted, the appropriate primary session is determined and the primary response is sent.

Related to the topic of distributed session management, is the establishment of TCP connections between *Retrieval Managers*. When a *CIP Client* initiates a session with a *Retrieval Manager*, a socket must be established over which all subsequent TCP packets are passed. Establishing a UNIX process for the socket takes time which delays accomplishing the distributed operation. To avoid having the delay associated with establishing a socket between *Retrieval Managers*, the TCP connection will be maintained between *Retrieval Managers* longer than an individual *CIP Session.* The *Retrieval Manager* to *Retrieval Manager* sessions will be multiplexed over two sockets which are not closed at the end of a *CIP Session*.

At the first instance of a *CIP Session* between two *Retrieval Managers*, two sockets are established. Each socket will handle message passing in a one-way direction: one for outgoing messages the other for incoming messages. The messages will be passed over the two TCP connections between the *Retrieval Managers* for that *CIP Session* and are maintained at the end of any particular *CIP Session*. Messages from multiple *User Sessions* are multiplexed over the single outbound and single return sockets.. The TCP Connections will only be closed if there have been no CIP messages for a period of time which will be determined by the two RMAs involved.

5.1.5 Directory Services

Key to the distributed three-tier architecture (Section 2) is the use of a directory service. As mentioned in Section 2, ICS uses a simple directory service for the conversion of high-level names to network address. This is similar to he type of services provided by the telephone companies' White Pages. Additional directory services are not currently used by ICS, e.g., but to resource location service, Yellow Pages services, mail address lookup. Based on these requirements and based on the choice of TCP/IP, the Domain Name Service (DNS) is used in ICS. Future enhancements to ICS may investigate the use of a yellow pages directory service listing of Retrieval Managers. This would augment the discovery function currently met using the Global Node.

To perform a search, a specific collection must be targeted in the search. The syntax which CIP requires for collection names follows a URL structure. The URL contains an Internet domain name for the Retrieval Manager on which the collection can be found. DNS contains information about the mapping of host and domain names, such as, "eos.nasa.gov", to IP addresses. This is done so that humans can use easily remembered names for machines rather than strings of numbers. It is maintained in a distributed fashion, with each DNS server providing name service for a limited number of domains. Also, secondary name servers can be identified for each domain, so that one unreachable network will not necessarily cut off name service.

An additional directory service is needed when a Retrieval Manager chooses to implement the Public Key Infrastructure for authentication. ICS uses the Lightweight Directory Access Protocol (LDAP) for interacting with a Certification Authority. LDAP is a TCP/IP version of the X.500 Directory Service which is defined by the ITU-T (formerly CCITT) [R23]. The ITU X.509 Directory - Authentication Framework defines authentication services are provided by the Directory to its users. The Directory can usefully be involved in meeting their needs for authentication and other security services because it is a natural place from which communicating parties can obtain authentication information of each other: knowledge which is the basis of authentication. The Directory is a natural place because it holds other information which is required for communication and obtained prior to communication taking place.

5.2 CEOS Network Connectivity

This section defines the connectivity between ICS *Retrieval Managers* that are provided by the CEOS Network. This section contains the following information:

- An overall CEOS Network Architecture
- Specific *Retrieval Manager* sites at CEOS Agencies
- Performance Requirements on CEOS Network based on user query requirements

5.2.1 CEOS Network Architecture

The CEOS Network architecture is shown in Figure 5-4. Within each participating agency or country, a *Retrieval Manager* is provided. It is the intent that the users in a country use that country's national Internet (or other national network resources) to access its local *Retrieval Manager*. To satisfy user requests, a *Retrieval Manager* may then need to access data from another participating *Retrieval Manager*. Two network alternatives are available to provide this access. One alternative is to use the worldwide Internet. While the worldwide Internet is quite ubiquitous, its performance is not dependably high.

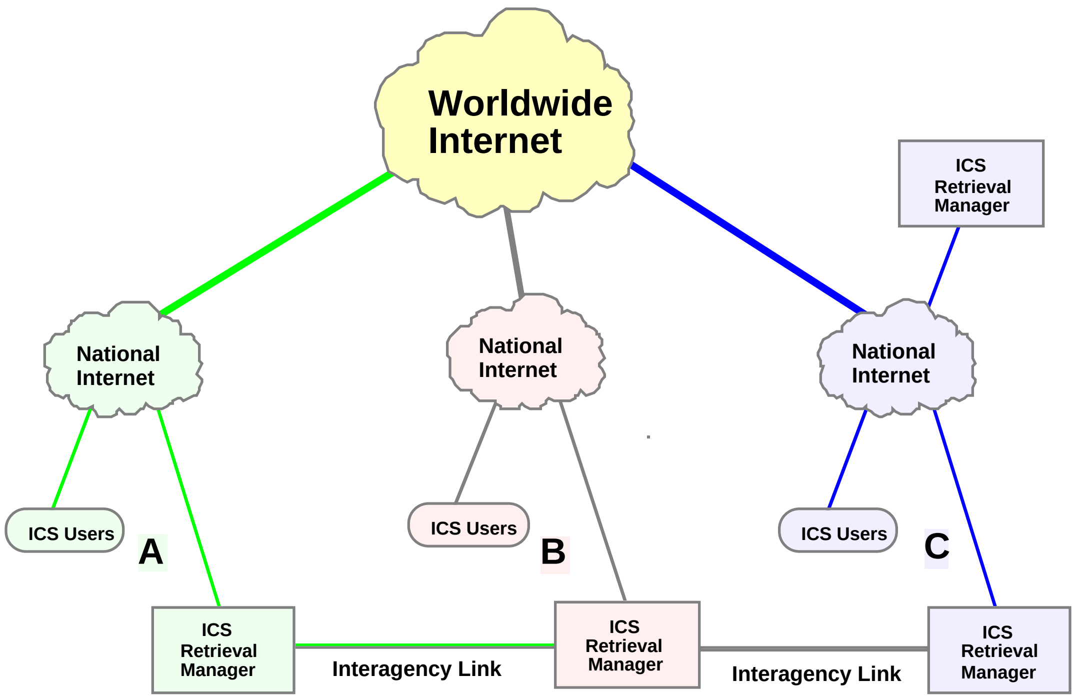


Figure 5-4. ICS Networks Model

On an individual, bi-lateral basis, the participating CEOS members may choose to implement private circuits between their facilities. These private circuits can be made available for access to *Retrieval Managers*, and may or may not also provide connectivity for other bilateral services. Collectively, these private circuits between participating CEOS organizations constitute the CEOS Network. Note that the CEOS Network is thus not a separate network consisting of a distinct set of circuits and equipment, but instead is a logical network of components provided by participants and used for CEOS purposes.

The nominal data flows to satisfy a user query is:

1) User accesses local *Retrieval Manager* via national (or agency) network resources

2) *Retrieval Manager* contacts cooperating *Retrieval Managers* via CEOS net (preferred) and/or the worldwide Internet as appropriate.

3) Cooperating *Retrieval Managers* access local database resources to formulate response.

4) Responses are returned to originating *Retrieval Manager* via same network which carried the request.

5) Originating *Retrieval Manager* collects the responses, and delivers them to the user via the national or agency infrastructure.

Note that it is not essential for an organization to have dedicated circuits to have a participating *Retrieval Manager*. Subject to performance limitations, the worldwide Internet provides the connectivity required. In Figure 5-4, dedicated links are shown (as an example only) between agencies A and B, and between B and C. In this example there is no dedicated link between A and C.

Note that technically in Figure 5-4, it is possible for the *Retrieval Manager* at A to access C via the pair of dedicated links (A-B and B-C). However, policy issues must be resolved by each agency for this to be allowed. Otherwise, the worldwide Internet can provide connectivity between A and C

The TCP/IP protocols will be used for communications between ICS *Retrieval Managers*. Interconnected routers in the CEOS Network must support network protocols which are robust and consistent with existent Internet protocols and standards (see RFC 1600 for applicable standards). All interconnecting networks attaching to CEOS Network are at least IP-based relays to other networks or user sites. Additional network protocols or enhancement features are allowed but should not affect the network availability, performance, or interconnectivity.

5.2.2 Organization and Physical Locations

CEOS Network provides network access to *Retrieval Managers* at the CEOS agency and affiliate sites listed in Table 5-1. Note that the sties listed in Table 5-1 are relatively permanent ICS sites, but due to the dynamic nature of the federation nature of ICS, this list will be dynamic.

Table 5-1. CEOS ICS Retrieval Manager Sites

Agency	Facility	Location	Key
BNSC	Natural Environment Research Council (NSRS)		f1
BNSC	National Remote Sensing Center (NRSC)		f2
CCRS	Canada Centre for Remote Sensing		f3
CSIRO	Commonwealth Scientific & Industrial Research Organization.		f4
DLR	Deutsche Forschungsanstalt fur Luft-und Raumfahrt		f5
ESA	European Space Research Institute (ESRIN)	Frascati, Italy	f6
EU/CEO			f7
Eumetsat			f8
EurImage			f9
IRE RAS	Inst. of Radio Eng. Russian Academy of Science		f10
NASA	Langley Research Center (LaRC) - TBR	Hampton, VA, USA	f11
NASDA	Earth Observation Center (EOC)	Hatoyama, Japan	f12
SpotImage			f13

Note: Although the CEOS Network provides connections to the CEOS agency facilities listed above, it is not the sole Internet provider to those sites.

5.2.3 ICS Requirements for CEOS Network

Performance of the CEOS Network will be critical to the ICS users satisfaction with distributed queries.

Table 5-2 provides estimates of the point–to–point data transfer requirements for each ICS *Retrieval Manager*. At present, there are no assumptions for peak utilization—the numbers in this table represent the average expected flow over a 24 hour period. It is anticipated that the CEOS Network Working Group will use the estimates in Table 5-2 along with other estimates for traffic on CEOS Network in order to estimate the overall bandwidth required for CEOS Network.

Also of importance to the overall performance of ICS is the Availability of CEOS Network links. In particular, each CEOS Network link shall have an operational availability of TBD (0.96) at a minimum and a Mean Down Time of TBD (4) hours or less.

Table 5-2. Network Load Characterization by ICS Site (Kbps)
(Some Data Entries To Be Supplied)

	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13
f1	–												
f2		–											
f3			–										
f4				–									
f5					–								
f6						–							
f7							–						
f8								–					
f9									–				
f10										–			
f11											–		
f12												–	
f13													–

The “fn” codes in Table 5-2 may be translated into their corresponding facilities by use of the “key” column in Table 5-1.

5.2.3.1 Approach to Characterization of Network Loading

The development of the equations used to populate Table 5-2 is detailed in the ICS Network Bandwidth Study [R20], only a brief outline of the development will be presented here.

There are five basic CIP services that were considered in the study: Initialization, Search, Present, Delete, and Terminate. If one can determine the number of times that each service is invoked during a period of time and knowing something about the size of each services message, one can estimate the network loading. Following the lead of the ICS Collection Technical Note [R11], the collections were modeled as a uniform tree (a type of directed graph). Use of trees allows the estimation of the number of messages passed over the CEOS network per service. Knowing the number of messages and estimating the number and type services requested over a period of time, one can calculate the network load.

Using this approach, one has to know or estimate a number of parameters. These parameters and their nominal values are listed in the next section.

In the course of developing the equations for the network load estimation, several assumptions had to be made. These assumptions are listed below.

[A1] The number of messages that will be passed between *Retrieval Managers* is proportional to the fraction of collection–to–collection links that point to a collection held by another *Retrieval Manager*. This is the “R” parameter listed in Table 5-3. As an example, if all collections were local, the R parameter would be zero (0). Likewise, if every collection pointed to a collection held by another *Retrieval Manager,* the R parameter would be one (1).

[A2] The probability that a specific *Retrieval Manager* is referenced by another *Retrieval Manager* is proportional to the portion of the total data holding held at that *Retrieval Manager*. The first assumption [A1] helps in the calculation of the number of network messages. It does not, however, give an indication to which *Retrieval Manager* the messages are directed. The best way to determine which *Retrieval Manager* will be referenced is to examine the collections and count which *Retrieval Managers* are referenced most often. Since that is not possible at this stage, this assumption allows a way to determine how messages are routed.

While not being an assumption directly, how the user load is modeled has significant impact on network loads. The user load was modeled for network loading purposes using the parameters Q (queries per user session), P (presents per user session), and Z (number of user sessions per unit of time).

5.2.3.2 Parameters Used for Network Load Estimation

Table 5-3 defines parameters used for developing the CEOS Network loading. Included in the table are the units and the nominal values used in the development of Table 5-2.

Table 5-3. Definition of Network Loading Parameters (1 of 2)
(Some Data Entries To Be Supplied)

Name	Value	Units	Definition
A		n/a	The average number of collections per collection
D		n/a	Collection depth (number of levels in collections)
$eff_{protocol}$	0.8	n/a	Efficiency of the protocol (number of data bits divided by the total number of bits used in the messages).
H_{col}		n/a	Average number of hits returned by a search of the collection descriptions.
H_{prod}		n/a	Average number of hits returned by a search of the product descriptions.
N_{prod}		n/a	Average number of product descriptions per terminal collection.

Table 5-3. Definition of Network Loading Parameters (2 of 2)
(Some Data Entries To Be Supplied)

Name	Value	Units	Definition
N_{RM}		n/a	Number of product and collections descriptions at a specified *Retrieval Manager*.
O_{col}		n/a	Probability that a collection will overlap another collection
O_{prod}		n/a	Probability that one product description will overlap with another.
$P_{col,RM}$		n/a	Average number of presents of collection descriptors performed at a specified *Retrieval Manager* during an average user session.
$P_{prod,RM}$		n/a	Average number of presents of product descriptors performed at a specified *Retrieval Manager* during an average user session.
Q_{RM}		n/a	Average number of searches performed at a specified *Retrieval Manager* during an average user session.
R		n/a	Ratio of the number of remote links to the total number of collection-to-collection links.
S_{col}		bytes	Average size of a collection descriptor
S_{delete}		bytes	Average size of delete-result-set message
S_{d_reply}		bytes	Average size of the response to a delete-result-set
S_{init}		bytes	Average size of an initialize command
S_{i_reply}		bytes	Average size of the response to an initialize
$S_{present}$		bytes	Average size of present message
S_{prod}		bytes	Average size of a product descriptor
S_{search}		bytes	Average size of search message
S_{s_reply}		bytes	Average size of the response to a search request
S_{term}		bytes	Average size of a terminate message
S_{t_reply}		bytes	Average size of the response to the terminate message
Z_{RM}		#/time	Average number of user sessions that occur over a given time at a specified *Retrieval Manager*.

5.2.3.3 Bandwidth for Browse

Another driver in the amount of CEOSnet bandwidth required to support ICS users is the retrieval of browse images. The CEOS Browse Task Team has begun to develop several parameters needed to support an estimate of the CEOSnet bandwidth necessary to support ICS user's request for browse images.

The first parameter in the Browse bandwidth estimation is the browse data size after compressed (to transfer via WAN). A maximum size is 1MB for EOS-HDF browse size with other agencies browse size is less than about 200 KB. So, an average browse size is estimated to be 500 KB

The next parameter to be estimated is the user's expected time for transfer of a browse image. Current performance for DLR and the University of Rhode Island provides image retrieval of from 45 to 60 seconds per browse data. The Browse Task Team suggests that a desirable requirement is 30 seconds.

The next parameter for estimating the bandwidth for a given ICS node is the frequency of browse retrieval requests. Given that CIP is a session oriented protocol, one approach to estimating the number of browse requests is to estimate the number of concurrent sessions. The number of sessions will have a diurnal variation, but only the peak number of sessions and the related peak number of requests are important for estimating the needed bandwidth. Also as the purpose of the worldwide ICS is realized, the diurnal variation of requests at a *Retrieval Manager* will be reduced as more users are retrieving data from around the globe. The ICS URD requires the *Retrieval Manager* be design to be capable of supporting a peak load of a minimum of 100 concurrent user interaction sessions. It is expected that all ICS data providers should support a minimum of 30 simultaneous sessions for a *Retrieval Manager*, this includes remote sessions to remote *Retrieval Managers*. The CEOS Browse Task Team estimates an average of 10 sessions. This number is 1/3 minimum multiple sessions requirement for ICS URD as some sessions of CIP are used for catalogue retrieving and ordering.

If we assume that this 1/3 fraction of users is continually executing browse requests these numbers provide the following network bandwidth requirement:

$$500 \text{ KB} * 10 \text{ sessions} / 30 \text{ sec} = 1333 \text{ Kbps} \approx 1.5 \text{ Mbps (T1)}$$

Of course, this estimate will change as the parameters are estimated by each agency. So, this bandwidth requirement is only a guideline.